PHAST--A Program for Simulating Ground-Water Flow and Multicomponent Geochemical Reactions

By David L. Parkhurst, Kenneth L. Kipp, Peter Engesgaard, and Scott R. Charlton

INTRODUCTION

PHAST is a 3-dimensional multicomponent reaction-transport model which can be used to simulate transient ground-water flow with or without geochemical reactions. The reaction-transport equations are solved by using the sequential approach (Yeh and Tripathi, 1989) in which transport and reaction are divided into separate calculations for each time step. First the components are transported and then geochemical reactions are calculated. No iterations between transport and reaction calculations within a time step are performed. The solute-transport simulator HST3D (Kipp, 1987, 1998) is used for the transport calculations and the program PHREEQC (Parkhurst, 1995, 1999) is used for the geochemical calculations.

The flow and transport part of PHAST uses porous media properties, initial conditions, and boundary conditions that are defined by zones for a point-distributed finite-difference grid. The types of boundary conditions include specified head, specified flux, and leaky boundary conditions; specialized boundary conditions for rivers and wells are included. The chemical composition at a boundary element is either fixed or a specified solution composition is associated with any inflow through the boundary. Either free surface or confined flow can be simulated. A flow or transport simulation comprises a series of one or more simulation periods during which boundary conditions are constant. The finite difference equations for flow and for transport of each chemical component are solved by the D4 Gaussian-elimination method or the restarted generalized conjugate-gradient method applied to a reduced matrix.

The geochemical part of the program has the capability to perform a wide range of chemical reaction calculations that include aqueous complexation, mineral equilibria, surface complexation, ion exchange, solid-solution equilibria, gas-phase equilibration, and general kinetic reactions. In addition, geochemical simulations, which include all of these types of reaction calculations plus mixing, irreversible reactions, and

temperature variation, may be used to define initial and chemical boundary conditions for the reaction-transport simulations. Essentially, any modeling capability available in PHREEQC may be used to establish initial and boundary conditions for the reactive-transport simulation of PHAST.

All spatially distributed properties are defined by zones that are rectangular prisms. Zones may overlap, in which case the order of definition is important because the last specification of a property for a cell or element will be the one used in the simulation. The units for input of properties may be a mixture of English and SI metric, however, all output data are SI metric with a user-selected time unit.

The transport part of PHAST is written in Fortran 90 and the geochemical part is written in C. Both parts dynamically allocate any computer memory necessary for program execution. Little effort has been expended to minimize storage requirements. Consequently, depending on the size of the domain, the program may require relatively large amounts of memory for execution.

In the application of PHAST, it is advisable to progress from simple to increasingly complicated simulations. First, the geochemical model PHREEQC should be used to evaluate chemical reactions by the use of inverse, reaction-path, and 1D reactive-transport calculations. Second, PHAST should be used in flow-only mode to obtain an acceptable model of the flow system. Finally, PHAST should be used for reaction-transport calculations that combine the flow simulation with geochemical reactions.

PHAST is now in the beta-testing phase, and the source code is not available for general release. The documentation of the input files has been written, but a complete manual containing the theory and example calculations is not yet available.

Executing PHAST

For reaction-transport calculations PHAST needs three data files for execution, (1) the flow and transport data file, (2) the chemical data file, and (3) the thermodynamic database file. For flow-only calculations, only the flow and transport data file is needed. Running PHAST is a two-step process that is automated in scripts that are provided for Unix and Windows. A prefix, from which input and output file names are derived, is defined as an argument to

the script. Within the script, the program PHASTINPUT is run to generate an intermediate input file named *Phast.tmp*. Following successful completion of the PHASTINPUT program, the script invokes the program PHAST, which performs the reaction-transport calculations. The script (Unix or Windows) is invoked as follows from a command-line window:

phast *prefix* [*database*] (The flow and transport data file must be named *prefix.***trans.dat**; the chemical data file must be named *prefix.***chem.dat**. Optionally, *database* is the name of the thermodynamic database file; if *database* is not defined, the database file must be named **phast.dat**.

If only flow is simulated, PHAST requires only the file *prefix.***trans.dat**; if solute transport and reactions are modeled in addition to flow, PHAST requires the files *prefix.***trans.dat**, *prefix.***chem.dat**, and the thermodynamic database file (*phast.dat*, by default).

Results of a PHAST run are written to a set of output files. In general, files with names including "*prefix***.O.**" are intended to be viewed with an editor. Files with names including "*prefix***.xyz.**" are files that may be imported to a spreadsheet or post-processed using data analysis programs. The file *prefix***.h5** is a hierarchical data format (HDF) file that is intended to be the standard repository for model results; this file is used by the 3D visualization program ModelViewer (Hsieh, 2002) and data may be extracted from the HDF file with the Java program **phast_hdf_export**. The complete list of output files is as follows:

*prefix***.log**--The *prefix***.trans.dat** and *prefix***.chem.dat** input data files are echoed to this file along with any error or warning messages related to processing the files. Where possible, error messages immediately follow the line from the input file that contains the error. Additional log messages indicate the progress of the simulation through stress periods and time steps. All error and warning messages are written to this file. The identifiers **-echo_input** in **PRINT_INITIAL** data block in the flow and transport data file and in the **PRINT** data block

in the chemistry data file allow echoing of the input data files to be disabled. The **-progress_statistics** identifier in **PRINT_FREQUENCY** data block of the flow and transport data file can be used to limit detailed printing of progress statistics to this file.

*prefix***.h5**--A Hierarchical Data Format (HDF) file containing grid and boundary condition information, heads, velocities, and selected chemical data. Grid node locations and boundary condition nodes are written to the HDF file. Boundary conditions include specified value, leaky, flux, river, and well boundary conditions. Heads and velocities are written at selected time intervals during the simulation as defined by the **-hdf_head** and **-hdf_velocity** identifiers in the **PRINT_FREQUENCY** data block and the **-hdf_head** and **-hdf_steady_flow_velocity** identifiers in the **PRINT_INITIAL** data block in the flow and transport input file. The chemical data to be written are defined in the **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file. These chemical data are written at selected time intervals during the simulation as defined by the **-hdf_chemistry** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

*prefix***.head.dat**--Heads at the final time step of the run are written to this file in a form that can be read as initial head conditions in subsequent runs. Writing to the file is controlled by the **-save_final_heads** identifiers of the **PRINT_FREQUENCY** data block in the flow and transport input file.

*prefix***.O.bal**--Fluid and chemical component global mass balances, total balances, and balances separated by boundary condition type are written to this file. These data are written at selected time intervals during the simulation as defined by the **-flow_balance** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

prefix.**O.bcf**--Fluid and solute flow rates through each boundary-condition cell are listed in this file by boundary condition type. These data are written at selected time intervals during the simulation as defined by the **-bc_flow_rates** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. Data in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

prefix.**O.chem**--Solution concentrations, distribution of aqueous species, saturation indices, and compositions of exchangers, gas phases, kinetic reactants, phase assemblages, solid solutions, and surfaces are written to this file during the internal PHREEQC calculation at the beginning of a PHAST run. By default, no results are written to this file during the transport calculations because a long printout will occur for every cell for every timestep, which could use all available disk space. However, for diagnostic purposes, for small problems, or for selected locations, it is possible to enable printing during the transport calculation by using **-force_chemistry_print** in the **PRINT_FREQUENCY** data block in the flow and transport data file. The **PRINT_LOCATIONS** data block in the flow and transport data file determines the cells for which data will be printed during the transport calculation. The **PRINT** and **USER_PRINT** data blocks in the chemistry input file can be used to specify the data that are written to *prefix*.**O.chem**.

prefix.**O.comps**--Static information and transient total dissolved concentrations for elements (components) are written to this file. Static data include the indices and mixing fractions for solutions, equilibrium phases, exchangers, surfaces, gas phases, solid solutions, and kinetic reactions that define initial conditions for the simulation. Static data and initial component concentrations are written at the beginning of the run as defined by the **-components** identifier in the **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-components** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. Data

in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

prefix.**O.head**--Static information and transient potentiometric head at cells are written to this file. Static data and initial heads are written at the beginning of the run as defined by the **-heads** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-heads** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. Data in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

prefix.**O.kd**--Static *X*, *Y*, and *Z* fluid conductances and solute dispersive conductances by cell face are written to this file. Transient conductance factors are also written to this file. Static data are written at the beginning of the run as defined by the **-media_properties** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-conductances** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. Data in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

prefix.**O.probdef**--Static and initial-condition flow and transport information are written to this file, including array sizes, grid definition, permeability distribution, porosity distribution, dispersivity distribution, fluid properties, initial head distribution, indices for initial conditions in each cell (solutions, pure phases, exchangers, kinetic reactions, solid solutions, surfaces, and gas phases), initial component concentrations in each cell, specific storage distribution, static and transient boundary-condition information, cell volumes, component concentrations for fixed and associated solutions for each boundary-condition cell. Much of

the data can be included or excluded from this file by options **-boundary_conditions**, **-fluid_properties**, and **-media_properties** in the **PRINT_INITIAL** data block of the flow and transport data file. The *prefix***.O.probdef** file is used to ensure the input data properly define the simulation. Additional transient information is written to the file according to the option **-boundary_conditions** in the **PRINT_FREQUENCY** data block of the flow and transport data file. Many of the data in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

*prefix***.O.vel**--Interstitial X-, Y-, and Z-velocities across cell faces and X-, Y-, and Z-velocities at grid nodes are written to this file. Steady-state flow velocities are written to this file as defined by the **-steady_flow_velocities** identifier in **PRINT_INITIAL** data block of the flow and transport data file, provided steady flow is simulated (**STEADY_FLOW** data block in the flow and transport data file). Transient velocities are written at selected time intervals during the simulation as defined by the **-velocity** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. Data in the file are formatted in 2D planes defined by **-print_orientation** in the **GRID** data block of the flow and transport data file.

*prefix***.O.wel**--Static and transient well information is written to this file, including well location and identification number, fluid and solute flow rates, cumulative fluid and solute flow amounts, and solute concentrations. Static data are written at the beginning of the run as defined by the **-wells** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-wells** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

*prefix***.xyz.chem**--Selected initial condition and transient chemical data for solutions, pure-phase assemblages, surface assemblages, exchange assemblages, kinetic reactions, solid solutions, and gas phases are written to this file. The data items to be written are defined in the

SELECTED_OUTPUT and **USER_PUNCH** data blocks of the chemical data file. Initial condition data are written at the beginning of the run as defined by the **-xyz_chemistry** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-xyz_chemistry** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file. The **PRINT_LOCATIONS** data block in the flow and transport data file specifies the cells for which data will be printed.

*prefix***.xyz.comps**--Initial condition and transient cell concentrations for each element (component) are written to this file for all cells in the sequence of increasing X, then Y, then Z. Initial condition data are written at the beginning of the run as defined by the **-components** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-components** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

*prefix***.xyz.head**--Initial condition and transient cell potentiometric heads are written to this file for all cells in the sequence of increasing X, then Y, then Z. Initial condition heads are written at the beginning of the run as defined by the **-heads** identifier in **PRINT_INITIAL** data block of the flow and transport data file. Transient data are written at selected time intervals during the simulation as defined by the **-heads** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

*prefix***.xyz.vel**--Steady-state or transient velocity-vector components located at grid nodes are written to this file for all cells in the sequence of increasing X, then Y, then Z. Steady-state flow velocities are written to this file as defined by the **-xyz_steady_flow_velocities** identifier in **PRINT_INITIAL** data block of the flow and transport data file, provided steady flow is simulated (**STEADY_FLOW** data block in the flow and transport data file). Transient

velocities are written at selected time intervals during the simulation as defined by the **-xyz_velocities** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

*prefix***.xyz.wel**--Transient concentration data for wells are written to this file. The concentration data are written at selected time intervals during the simulation as defined by the **-xyz_wells** identifier in the **PRINT_FREQUENCY** data block of the flow and transport data file.

selected_output--Selected chemical data for solutions, pure-phase assemblages, surface assemblages, exchange assemblages, kinetic reactions, solid solutions, and gas phases are written to this file only during the internal call to PHREEQC to process the chemical data file. The file name for this file is defined with the **-file** identifier in the **SELECTED_OUTPUT** data block and data items to be written are defined in the **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file.

Purpose and Scope

The purpose of this report is to describe the use of the three-dimensional reaction-transport model PHAST. The scope includes outlining the flow, transport, and reaction capabilities of the program, describing all data necessary for the transport data file, and presenting an example that uses many of the capabilities of the program.

Data entry for the chemical data file and thermodynamic database file is identical to the input for the program PHREEQC. Data input for PHREEQC is described in Parkhurst and Appelo (1999) and the user is referred to this documentation, which is available at the web sites *http://wwwbrr.cr.usgs.gov/projects/GWC_coupled/phreeqc* or *http://water.usgs.gov/software*.

DESCRIPTION OF DATA INPUT

The input for PHAST is divided into three files: (1) a flow and transport data file that contains the necessary data for flow and transport calculations, including definitions of media properties, initial conditions, and boundary conditions; (2) a chemical data file that contains all of the chemical information that are used for initial conditions

and boundary conditions including initial- and boundary-condition solution compositions, and initial conditions for kinetic reactions, pure-phase assemblages, exchange assemblages, solid-solution assemblages, surface complexation assemblages, and gas phase compositions that are present in each cell; and (3) a thermodynamic database file that contains the definition of all chemical elements including rate expressions for kinetic reactions and stoichiometric and thermodynamic data for each aqueous, exchange, and surface species, and each mineral or gas component. The data in thermodynamic database file can be augmented, overridden, or included entirely in the chemical data file. Initial and boundary conditions are selected in the flow and transport data file using index numbers that refer to numbered sets of kinetic reactants, and numbered solution, pure-phase-assemblage, exchange-assemblage, solid-solution-assemblage, surface-assemblage, and gas-phase compositions that are defined in the chemical data file. The data in each file are arranged by keyword data blocks. Each data block begins with a line that contains the keyword (and possibly additional data) and usually is followed by additional lines containing data related to the keyword.

The following keywords are used in the flow and transport data file (listed alphabetically): **CHEMISTRY_IC**, **END**, **FLOW_ONLY**, **FLUID_PROPERTIES**, **FLUX_BC**, **FREE_SURFACE_BC**, **GRID**, **HEAD_IC**, **LEAKY_BC**, **MEDIA**, **PRINT_FREQUENCY**, **PRINT_INITIAL**, **PRINT_LOCATIONS**, **RIVER**, **SOLUTION_METHOD**, **SPECIFIED_VALUE_BC**, **STEADY_FLOW**, **TIME_CONTROL**, **TITLE**, **UNITS**, and **WELL**. Simulation periods are defined by a set of keyword data blocks and conclude with an **END** keyword. The flow and transport data file contains all of the flow parameters, initial conditions, and boundary conditions necessary for one or more simulation periods. The flow and transport data file is read first during a PHASTINPUT run. As it is processed, the data for each simulation period are checked for completeness and consistency, then an intermediate flow and transport data file that is similar to an HST3D input file is written. When the program PHAST is run, the thermodynamic database file and the chemical data file are read and initial geochemical calculations are performed. After PHAST completes the initial geochemical calculations, the reaction and transport calculations are performed, one simulation period at a time.

The following keywords are used in the chemical data file and the thermodynamic data base file (listed alphabetically): **ADVECTION**, **END**, **EQUILIBRIUM_PHASES**, **EXCHANGE**, **EXCHANGE_MASTER_SPECIES**, **EXCHANGE_SPECIES**, **GAS_PHASE**, **INCREMENTAL_REACTIONS**, **INVERSE_MODELING**, **KINETICS**, **KNOBS**, **MIX**, **PHASES**, **PRINT**, **RATES**, **REACTION**, **REACTION_TEMPERATURE**, **SAVE**, **SELECTED_OUTPUT**, **SOLID_SOLUTIONS**, **SOLUTION**, **SOLUTION_MASTER_SPECIES**, **SOLUTION_SPECIES**, **SOLUTION_SPREAD**, **SURFACE**, **SURFACE_MASTER_SPECIES**, **SURFACE_SPECIES**, **TITLE**, **TRANSPORT**, **USER**, **USER_PRINT**, and **USER_PUNCH.** These data blocks include all keywords for the program PHREEQC and any PHREEQC calculation can be accomplished with PHAST to produce initial and boundary conditions for flow and transport. Data are read from the chemical data file until an **END** keyword is encountered, after which the specified geochemical calculations are performed. The process of reading data from the chemical data file until an **END** is encountered followed by performing calculations is repeated until the end of the chemical data file is encountered. When the processing of the chemical data file is completed all of the chemical compositions needed for initial and boundary conditions of the multicomponent reaction-transport simulation must have been defined, including all initial and boundary conditions solution compositions, and all initial conditions for the phase-assemblage, exchange-assemblage, solid-solution assemblage, surface-assemblage, and gas-phase composition in each active cell.

Conventions for Data Input

PHAST uses modular keyword data blocks to enter data.The program is insensitive to the order of keyword data blocks in the thermodynamic database file and between **END** keywords in the chemical data file. For the flow and transport data file, the order of the keyword data blocks is important for boundary conditions and initial conditions because element and cell parameters may be defined multiple times, for example, by overlapping zones; however, only the final definition of a property for an element or cell is used in the calculations.

Data for the program are free format; spaces or tabs may be used to delimit input fields. As much as possible, the program is case insensitive. The important exception to this rule regards chemical formulas, which are case dependent. Data elements entered on a single line are order specific.

The following conventions are used for all data files for PHAST.

Keywords--Input data blocks are identified with an initial keyword. This word must be spelled exactly, although case is not important. Several of the keywords have synonyms. For example **PURE_PHASES** is a synonym for **EQUILIBRIUM_PHASES**.

Identifiers--Identifiers are options that may be used within a keyword data block. Identifiers may have two forms: (1) they may be spelled completely and exactly (case insensitive) or (2) they may be preceded by a hyphen and then enough characters to uniquely define the identifier. The form with the hyphen is always acceptable. Usually, the form without the hyphen is acceptable, but in some cases the hyphen is needed to indicate the word is an identifier rather than an identically spelled keyword; these cases are noted in the definition of the identifiers in the following sections. In this report, the hyphen is usually used except for identifiers of the **SOLUTION** keyword and the identifiers **log_k** and **delta_h**. The hyphens are not used in these cases to avoid confusion about negative quantities. The hyphen in the identifier never implies the negative of a quantity is entered. For example, the identifier "**-log_k**" does not mean the negative of the log K, it is simply an alternate form for the identifier "**log_k**".

The order of identifiers is only important for definition of zones. Identifiers that define properties apply to the zone defined by the last **-zone** identifier.

Comments--The "#" character delimits the beginning of a comment in the input file. All characters in the line which follow this character are ignored. If the entire line is a comment, the line is not echoed to the output file. If the comment follows input data on a line, the entire line, including the comment, is echoed to the output file. The "#" is useful for adding comments explaining the source of various data or describing the problem set up. In addition, it is useful for temporarily removing lines from an input file.

Logical line separator--A semicolon (";") is interpreted as a logical end of line character. This allows multiple logical lines to be entered on the same physical line. For example, solution data could be entered as:

"pH 7.0; pe 4.0; temp 25.0",

on one line. The semicolon should not be used in character fields, such as the title or other comment or description fields.

Logical line continuation--A backslash ("\") followed by an end-of-line, causes the input processor to ignore the end-of-line. By ignoring the end-of-line character single logical line can be written to two physical lines. For example, a long chemical equation could be entered as:

"Ca0.165Al2.33Si3.67O10(OH)2 + 12 H2O = \"

"0.165Ca+2 + 2.33 Al(OH)4- + 3.67 H4SiO4 + 2 H+"

on two lines. The program would interpret this sequence as a balanced equation entered on a single logical line. Note that the backslash must directly precede the end-of-line; if a space or tab follows the backslash and precedes the end-of-line, the end-of-line will be interpreted as normal.

Formatting Conventions for Documentation

Several formatting conventions are used to help the user interpret the input requirements. Keywords are always capitalized and bold. Words in bold must be included literally when creating input data sets (although upper and lower case are interchangeable and optional spellings may be permitted). "Identifiers" are additional keywords that apply only within a given keyword data block; they can be described as sub-keywords. "**Temperature**" is an identifier for **SOLUTION** input. Each identifier may have one of two forms: (1) the identifier word spelled exactly (for example, "**temperature**"), or (2) a hyphen followed by a sufficient number of characters to define the identifier uniquely (for example, **-t** for temperature). Words in *italics* are input values that are variable and depend on user selection of appropriate values. Items in brackets ([]) are optional input fields. Mutually exclusive input fields are enclosed in parentheses and separated by the word "or". In general, the optional fields must be entered in the specified order. For clarity, commas are used to delimit input fields in the explanations of data input; however, commas are not allowed in the input data file; only white space (spaces and tabs) may be used to delimit fields in input data sets. Where applicable, default values for input fields are stated.

All line numbers associated with example input in the documentation are for identification purposes only. Data files for use with PHAST do not contain any line numbering.

The Flow and Transport Data File

The following sections describe the data input requirements for the flow and transport data file. This file includes all the data for flow and transport calculations, including media properties, initial and boundary conditions, solver selection and parameters, and print-frequency control. All data are input through keyword data blocks.

Spatial Data

The coordinate system for PHAST is three dimensional; *X* and *Y* represent horizontal axes and *Z* represents the vertical axis. The relation of *X* to *Y* is right-handed, in the sense that if your thumb is pointed in the positive *Z* direction, then your fingers curl from the positive *X* axis to the positive *Y* axis. The coordinate grid is defined by a set of node points, by which two sets of rectangular prisms, called cells and elements, are defined. Both the set of cells and the set of elements fill the entire domain of the simulation. A cell surrounds each of the nodes, extending halfway to each adjacent node or to the boundary of the domain. An element is a rectangular prism defined by the eight nodes that are located at its corners. Media properties, including porosity, hydraulic conductivity, specific storage, dispersivity, and active or inactive zones, are defined by element. All other properties, including initial conditions (heads, solution compositions, sets of kinetic reactants, and exchange, solid-solution, surface, and pure-phase assemblages) and boundary conditions (specified head and solution composition, specified flux and solution composition, and, for leaky boundaries, thickness, head on the distal side, and solution composition) are defined by cell.

Rivers are defined by a series of *X*-*Y* points, width, head, and additional data to define river leakance. Wells are defined by X-Y location, flow or injection rate, and additional data to define well characteristics and open intervals.

Most of the data in the flow and transport data file are spatially distributed properties of the medium and initial and boundary conditions. All spatially distributed data are input by rectangular prisms, called "zones", which are defined by two points in order: the left-front-lower corner and the right-back-upper corner. For each zone, a spatially distributed property may be entered by any of four methods: (1) a single value may be applied to each cell or element that falls within the zone, (2) a list of values, including exactly one for each cell or element that falls within the zone (this method is indicated by a word that begins with the letter "b", as in **by_cell**), (3) a list of values may be read from a file, including exactly one for each cell or element that falls within the zone (this method is indicated by a word that begins with the letter "f", as in **file**), or (4) values for the property are linearly interpolated along a given coordinate axis using property values at two points along the given axis (this method is indicated by a coordinate direction, **X**, **Y,** or **Z**). When method 2 or 3 is used, repeat factors can be used to input multiple data with the same values; for example, 11*2.6, in a list of values is interpreted as 11 values of 2.6. An asterisk must be used to indicate a repeat factor and no intervening spaces are allowed. Method 4 is used to distribute properties to cells, in which case, the node location is used for interpolation, and to elements, in which case the element centroid is used for interpolation. If the node or centroid falls outside the range of locations defined by the end points of the interpolation region, values from the closest end point are applied to the cell or element. When method 4 is used for any chemical property, such as compositions of solutions, kinetic reactions, or pure-phase assemblages, the compositions of the end members are mixed proportionally to the distance between the end points of a given cell or element along the specified axis.

As spatial data for a simulation period are entered, it is permissible to use overlapping zones. For example, the initial condition head for a given cell could be included in multiple zones. The initial head that is used in the flow or reaction-transport calculation will be the head from the last zone in the input file that defined the head in that cell. Frequently, it is convenient to define a uniform value for a property by using a zone that includes the entire domain. Additional zones can then be defined that overlay different property values in different parts of the domain to obtain a complete spatial definition of the property.

Keywords for the Flow and Transport Data File

The following sections describe the data input requirements for the flow and transport data file. Each type of data is defined with a specific keyword data block. The keyword data blocks are discussed in alphabetical order.

CHEMISTRY_IC

This keyword is used to define the initial conditions in the domain, including initial solution and optionally kinetic reactions; initial phase-, exchange-, and surface-assemblage compositions; and rarely initial gas-phase composition (usually gases are defined as fixed partial pressures in **EQUILIBRIUM_PHASES**). The compositions are defined by index numbers that refer to compositions defined with **SOLUTION**, **EQUILIBRIUM_PHASES**, **EXCHANGE**, **KINETICS**, **SOLID_SOLUTION**, **SURFACE**, and **GAS_PHASE** keywords or through initial geochemical calculations defined in the chemical data file that are subsequently saved with a **SAVE** data block. The **CHEMISTRY_IC** keyword data block is mandatory for all reaction-transport calculations.

Example

```
Line 0:  CHEMISTRY_IC
Line 1:        -zone      0     0     0     10    10    10
Line 2:             -solution  1
Line 3:             -equilibrium_phases   by_cell
Line 3a:                 1 2 1 3 1 1 3 60*2
Line 4:             -surface   file surface.fil
Line 5:             -exchange  X     1     0.0   5     10.0
Line 6:             -gas_phase 2
Line 7:             -solid_solutions      5
Line 8:             -kinetics  2
```

Explanation

Line 0: **CHEMISTRY_IC**

CHEMISTRY_IC is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone for initial conditions is defined. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone.

Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-solution** *property*

-solution--Initial solution index numbers are defined for the zone. Optionally, **solution** or **-s**[**olution**].

property--Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Solutions or mixtures of solutions may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell in the zone, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 2 illustrates method 1.

Line 3: **-equilibrium_phases** *property*

-equilibrium_phases--Initial pure-phase-assemblage index numbers are defined for the zone. Optionally, **equilibrium_phases**, **-eq**[**uilibrium_phases**], **pure_phases**, **-p**[**ure_phases**], **phases**, or **-p**[**hases**].

property--Pure-phase-assemblage compositions for the zone are based on index numbers that refer to pure-phase-assemblage compositions defined in the chemical data file. Pure-phase assemblages or mixtures of pure-phase assemblages may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell in the zone, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$,

distance$_2$, a composition is linearly interpolated from the end-member compositions defined by indices *index*$_1$ and *index*$_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between *distance*$_1$ and *distance*$_2$ along the given coordinate direction; for distances outside the range of *distance*$_1$ to *distance*$_2$, the composition at the nearest end point (*distance*$_1$ or *distance*$_2$) is used. Lines 3 and 3a use method 2. Note the use of the repeat factor in line 3a.

Line 4: **-surface** *property*

-surface--Initial surface-assemblage index numbers are defined for the zone. Optionally, **surface** or **-su**[**rface**].

property--Surface-assemblage compositions for the zone are based on index numbers that refer to surface-assemblage compositions defined in the chemical data file. Surface assemblages or mixtures of surface assemblages may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell in the zone, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, *index*$_1$, *distance*$_1$, *index*$_2$, *distance*$_2$, a composition is linearly interpolated from the end-member compositions defined by indices *index*$_1$ and *index*$_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between *distance*$_1$ and *distance*$_2$ along the given coordinate direction; for distances outside the range of *distance*$_1$ to *distance*$_2$, the composition at the nearest end point (*distance*$_1$ or *distance*$_2$) is used. Line 4 illustrates method 3.

Line 5: **-exchange** *property*

-exchange--Initial exchange-assemblage index numbers are defined for the zone. Optionally, **exchange** or **-ex**[**change**].

property--Exchange-assemblage compositions for the zone are based on index numbers that refer to exchange-assemblage compositions defined in the chemical data file. Exchange assemblages or mixtures of exchange assemblages may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 5 illustrates method 4.

Line 6: **-gas_phase** *property*

-gas_phase--Initial gas-phase index numbers are defined for the zone. Optionally, **gas_phase** or **-g**[**as_phase**].

property--Gas-phase compositions for the zone are based on index numbers that refer to gas-phase compositions defined in the chemical data file. Gas phases or mixtures of gas phases may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$

and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 6 illustrates method 1.

Line 7: **-solid_solutions** *property*

-solid_solutions--Initial solid-solution-assemblage index numbers are defined for the zone. Optionally, **solid_solution**, **solid_solutions** or **-soli**[**d_solutions**].

property--Solid-solution-assemblage compositions for the zone are based on index numbers that refer to solid-solution-assemblage compositions defined in the chemical data file. Solid-solution assemblages or mixtures of solid-solution assemblages may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 7 illustrates method 1.

Line 8: **-kinetics** *property*

-kinetics--Kinetic-reaction-set index numbers are defined for the zone. Optionally, **kinetics** or **-k**[**inetics**].

property--Kinetic reactions for the zone are based on index numbers that refer to kinetic-reaction sets defined in the chemical data file. Kinetic-reaction sets or mixtures of kinetic-reaction sets may be entered for the zone with any of the four methods for defining a spatially distributed property:

(1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 8 illustrates method 1.

Notes

CHEMISTRY_IC defines initial conditions and, as such, the data block must be present in the data for the first simulation period, that is, the data block must occur before the first **END** keyword in the flow and transport data file. All index numbers must refer to index numbers for solutions, sets of kinetic reactants, equilibrium-phase assemblages, exchange assemblages, solid-solution assemblages, surface assemblages, and gas-phases that are defined in the chemical data file. Initial solution definitions are mandatory for all active cells in the domain. Sets of kinetic reactants, gas phases, equilibrium phases, and exchange, solid-solution, and surface assemblages are optional. A negative index number for an entity is interpreted to mean that entity is absent in the zone. By default, all entities (solution, set of kinetic reactants, assemblages, and gas phase) are absent from all cells. Multiple zones may be used within the **CHEMISTRY_IC** data block to define the initial conditions for the domain. The index number for a property for a single cell may be defined multiple times as part of different zone definitions. The index number used in the reaction-transport calculations for that entity for that cell is the last index number defined for it in the flow and transport data file.

If property input method 4, interpolation, is used to define compositions for cells, the composition of each cell is determined by linearly mixing the end-member compositions represented by two index numbers. The

mixing fractions are of the end members are determined by the location of the node for a cell relative to the locations specified for the end members along the given coordinate axis. For each type of initial condition the following are linearly mixed: for solutions, moles of each element and temperature; for sets of kinetic reactants, the moles of each reactant; for pure-phase assemblages, moles of each mineral phase; for exchangers, moles of exchange sites and moles of exchanged ions; for solid-solutions, moles of each component of each solid solution; for surfaces, moles of surface sites, surface area, and moles of sorbed elements; and for gas phases, moles of each gas component.

END

This keyword has no associated data. It ends the data input for a simulation period. Data for additional simulation periods may follow in the input data set, each simulation in turn is terminated with an **END** keyword.

FLOW_ONLY

This keyword is used to eliminate all transport and reaction calculations; only flow is simulated. Any keyword data blocks or identifiers related to transport or chemistry in the flow and transport data file are ignored. If the **FLOW_ONLY** data block is not included, the simulation is assumed to be a reaction-transport calculation.

Example

```
Line 0:  FLOW_ONLY        true
```

Explanation

Line 0: **FLOW_ONLY** [***True*** *or* ***False***]

FLOW_ONLY--switches the program from a reaction-transport model to a flow model. If a **FLOW_ONLY** data block is not included in the flow and transport data file, the simulation is assumed to be both flow and transport and reaction.

[***True*** *or* ***False***]--a value of **true** (optionally, **t**[**rue**]) indicates that only flow is simulated and no transport or reaction calculations are made. A value of **false** (optionally, **f**[**alse**]) indicates that transport and reaction calculations are made. If neither **true** nor **false** are entered, **true** is assumed.

FLUID_PROPERTIES

Fluid properties are scalar properties that are uniform over the entire domain, they are not spatially distributed. Required units are given in the example; alternative units are not allowed. If the **FLUID_PROPERTIES** data block is not included the default values listed in the example are used.

Example

```
Line 0:  FLUID_PROPERTIES
Line 1:        -compressibility      4.7e-10    # 1/Pa
Line 2:        -density              1000.0     # kg/m^3
Line 3:        -diffusivity          1e-9       # m^2/s
Line 4:        -viscosity            0.00115    # Pa-sec
```

Explanation

Line 0: **FLUID_PROPERTIES**

FLUID_PROPERTIES is the keyword for the data block, no other data are included on this line.

Line 1: **-compressibility** *compressibility*

-compressibility--Compressibility of the fluid is entered on this line. Optionally, **compressibility** or **-c**[**ompressibility**].

compressibility--Compressibility of the fluid in units of 1/Pascal. Default, 4.7×10^{-10} 1/Pascal.

Line 2: **-density** *density*

-density--Density of the fluid is entered on this line. Optionally, **density** or **-de**[**nsity**].

density--Density of the fluid in units of kg/m^3. Default, 1000 kg/m^3.

Line 3: **-diffusivity** *diffusivity*

-diffusivity--Molecular diffusivity is entered on this line. Optionally, **diffusivity** or **-di**[**ffusivity**].

diffusivity--Molecular diffusivity, which applies to all components, in units of m^2/sec. Default, 10^{-9} m^2/sec.

Line 4: **-viscosity** *viscosity*

-viscosity--Viscosity of the fluid is entered on this line. Optionally, **viscosity** or **-v**[**iscosity**].

viscosity--Viscosity of the fluid in units of Pascal-sec. Default, 0.00115 Pa-sec.

Notes

The units for each of the fluid properties are fixed; no mechanism for alternative units for these properties is provided.

If a free surface water table is modeled (**FREE_SURFACE** data block) then the compressibilities of the fluid and the matrix are automatically set to zero. If a free surface is not modeled, the compressibility of the matrix is calculated from the specific storage for each cell. If the specific storage is too small relative to the porosity and fluid compressibility, it is possible that the compressibility of the matrix will be calculated to be negative, which results in an error message and termination of the program. The resolution of this problem is to use a larger specific storage or to decrease the fluid compressibility to a value that is compatible with the specific storage.

FLUX_BC

This keyword is used to define the specified flux boundary conditions. For flow-only simulations, only the fluid flux for each cell with a specified flux boundary condition is required. For reaction-transport calculations, the index number of an associated solution is also required. This keyword data block is not needed if no specified flux boundaries are to be included in the simulation.

Example

```
Line 0:  FLUX_BC
Line 1:        -zone      0     10    10    10    10    10
Line 2:              -flux      -3.5e-3
Line 3:              -associated_solution  5
Line 4:              -face      Z
```

Explanation

Line 0: **FLUX_BC**

FLUX_BC is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone, which may be a single point (including 1 cell) or a 1- or 2-dimensional zone for specified flux boundary-condition definition. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone.

Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-flux** *property*

-flux--Volumetric fluxes are defined for the zone. Optionally, **flux** or **-fl**[**ux**].

property--The volumetric flux (L/T) for the zone may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly

interpolated from the end-point values defined by *value*$_1$ and *value*$_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between *distance*$_1$ and *distance*$_2$ along the given coordinate direction; for locations outside the range of *distance*$_1$ to *distance*$_2$, the value at the nearest end point (*distance*$_1$ or *distance*$_2$) is used. Line 2 illustrates method 1. The flux is a signed quantity that indicates whether the flux is in the positive coordinate direction or the negative coordinate direction. Units, L/T, are defined by **-flux** identifier in the **UNITS** data block.

Line 3: **-associated_solution** *property*

-associated_solution--Solution index numbers associated with the cells with specified flux boundary conditions. Optionally, **associated_solution**, **-a**[**ssociated_solution**], **solution**, or **-s**[**olution**].

property--Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Solutions or mixtures of solutions may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, *index*$_1$, *distance*$_1$, *index*$_2$, *distance*$_2$, a composition is linearly interpolated from the end-member compositions defined by indices *index*$_1$ and *index*$_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between *distance*$_1$ and *distance*$_2$ along the given coordinate direction; for distances outside the range of *distance*$_1$ to *distance*$_2$, the composition at the nearest end point (*distance*$_1$ or *distance*$_2$) is used. Line 3 illustrates method 1.

Line 4: **-face** (**X**, **Y**, or **Z**)

-face--If the zone is 0- or 1-dimensional, the coordinate direction of the flux must be specified.

Optionally, **face** or **-fa**[**ce**].

X, **Y**, or **Z**--The coordinate direction of the flux.

Notes

The zones for all boundary conditions must be 0-, 1-, or 2-dimensional. For specified flux boundary conditions with 2-dimensional zones, the coordinate direction of the flux can be determined from the zone definition and the **-face** identifier is not necessary. For 0- and 1-dimensional zones the coordinate direction of the flux is ambiguous, and the **-face** identifier must be used to identify the coordinate direction of the flux. The sign of the flux quantity indicates whether the flux is in the positive or negative coordinate direction. Note that the hyphen in **-flux** does not indicate a negative quantity; it only indicates that the word is an identifier.

Multiple zones may be used within any **FLUX_BC**, **LEAKY_BC**, or **SPECIFIED_VALUE_BC** data block to define boundary conditions within the domain. Different boundary conditions for a single cell (node) may be defined multiple times as part of different zone definitions and different keyword data blocks. The boundary condition that is used for a cell is the last zone definition in the flow and transport data file that defines a boundary condition for that cell.

FREE_SURFACE_BC

This keyword is used to define the presence of a free surface, or unconfined flow, for the calculations. If the **FREE_SURFACE** data block is not included, the simulation is assumed not to have a free surface, that is, confined flow is simulated.

Example

```
Line 0:  FREE_SURFACE_BC        true
```

Explanation

Line 0: **FREE_SURFACE_BC** [***True*** *or* ***False***]

FREE_SURFACE_BC--Keyword data block includes or excludes a free surface in the simulation. Optionally, **FREE_SURFACE**.

[***True*** *or* ***False***]--A value of **true** (optionally, **t**[**rue**]) indicates that a free surface, or unconfined flow, is simulated. A value of **false** (optionally, **f**[**alse**]) indicates that no free surface, that is, confined flow is simulated. If neither **true** nor **false** is entered, **true** is assumed. At the beginning of the run, the value is set to **false**.

Notes

This option is used to simulate confined (**FREE_SURFACE_BC false**) or unconfined (**FREE_SURFACE_BC true**) flow. If unconfined flow is simulated, the specific storage in each cell is automatically set to zero, which is equivalent to setting the fluid and matrix compressibilities to zero.

GRID

This keyword is used to define the grid range and spacing for the simulation domain. This keyword data block is mandatory for all simulations.

Example

```
Line 0:  GRID
Line 1:        -uniform   X     0.    1000. 6
Line 2:        -nonuniform      Y     0.    100.
Line 3:             400.  800.  1000.
Line 1a:       -uniform   Z     0.    10.   2
Line 4:        -overlay_uniform Z 0.0 1.0 11
Line 5:        -overlay_nonuniform Z 8.2 8.4 8.5 8.6 8.8
Line 6:        -snap Z  0.05
Line 7:        -chemistry_dimensions       XZ
Line 8:        -print_orientation          XZ
```

Explanation

Line 0: **GRID**

GRID is the keyword for the data block, no other data are included on this line.

Line 1: **-uniform** (**X**, **Y**, or **Z**) *minimum*, *maximum*, *number of nodes*

-uniform--Uniform grid spacing is defined for the given coordinate. **-Nonuniform** and **-uniform** are mutually exclusive for a given coordinate direction. Optionally, **uniform** or **-u**[**niform**].

X, **Y**, or **Z**--The coordinate direction for which uniform grid spacing is to be defined.

minimum--Minimum coordinate value for the domain in the given coordinate direction, unit is L. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block.

maximum--Maximum coordinate value for the domain in the given coordinate direction, unit is L. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block.

number of nodes--Number of nodes for the domain in the given coordinate direction. The number of elements in this direction is one less than the number of nodes. The number of cells in this direction is equal to the number of nodes.

Line 2: **-nonuniform** (**X**, **Y**, or **Z**) *list of node coordinates*

-nonuniform--Nonuniform grid spacing is defined for the given coordinate. **-Nonuniform** and **-uniform** are mutually exclusive for a given coordinate direction. Optionally, **nonuniform** or **-n**[**onuniform**].

X, **Y**, or **Z**--The coordinate direction for which nonuniform grid spacing is to be defined.

list of node coordinates--List of coordinates for the domain in the given coordinate direction, unit is L. List must be in ascending order. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block. List of nodes may continue on next line as shown in line 3.

Line 3: *list of node coordinates*

list of node coordinates--Continuation of list of node coordinates for nonuniform mesh as defined by last **-nonuniform identifier**.

Line 4: **-overlay_uniform** (**X**, **Y**, or **Z**) *minimum*, *maximum*, *number of nodes*

-overlay_uniform--Additional uniform grid spacing is defined for the given coordinate. Either **-nonuniform** or **-uniform** is required for each coordinate direction; optional **-overlay_uniform** and **-overlay_nonuniform** identifiers may be defined for each coordinate direction. Multiple overlays may define the same node or nodes that are close together; nodes within the *snap_distance* for the coordinate will be merged into a single node. Optionally, **overlay_uniform** or **-o**[**verlay_uniform**].

X, **Y**, or **Z**--The coordinate direction for which additional uniform grid spacing is to be defined.

minimum--Minimum coordinate value for the additional nodes in the given coordinate direction, unit is L. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block.

maximum--Minimum coordinate for the additional nodes in the given coordinate direction, unit is L. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block.

number of nodes--Number of additional nodes for the given coordinate direction.

Line 4 indicates that beginning with a node at 0.0 and ending with a node at 1.0, 11 equally spaced nodes will be added to the mesh in the Z direction.

Line 5: **-overlay_nonuniform** (**X**, **Y**, or **Z**) *list of node coordinates*

-overlay_nonuniform--Additional nonuniform grid spacing is defined for the given coordinate. Either **-nonuniform** or **-uniform** is required for each coordinate direction; optional **-overlay_uniform** and **-overlay_nonuniform** identifiers may be defined for each coordinate direction. Multiple overlays may define the same node or nodes that are close together; nodes within the *snap_distance* for the coordinate will be merged into a single node. Optionally, **overlay_nonuniform** or **-overlay_n**[**onuniform**].

X, **Y**, or **Z**--The coordinate direction for which additional grid nodes are to be defined.

list of node coordinates--List of one or more node coordinates in the given coordinate direction, unit is L. List must be in ascending order. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block. List of nodes may continue on next line.

Line 5 indicates that additional nodes are placed at 8.2, 8.4, 8.5, 8.6, and 8.7 in the Z direction.

Line 6: **-snap** (**X**, **Y**, or **Z**) *snap_distance*

-snap--Identifier to define the minimum distance between grid points. Optionally, **snap** or **-s**[**nap**].

X, **Y**, or **Z**--The coordinate direction for which *snap_distance* is to be defined.

snap_distance--Minimum distance between nodes in the given coordinate direction, unit is L. After coordinates are combined and sorted for a coordinate direction, nodes within this distance of the previous node will be eliminated. Units, L, are defined by **-horizontal_grid** and **-vertical_grid** identifiers in the **UNITS** data block. Default is 0.001.

Line 7: **-chemistry_dimensions** [**X**] [**Y**] [**Z**]

-chemistry_dimensions--If, conceptually, the transport calculation is 1 or 2 dimensional, the geochemical calculations can be performed in a single line or plane of cells and the resulting

chemical compositions can be replicated for the remaining lines or plane, thus, saving significant amounts of computation time. Optionally, **chemistry_dimensions**, **transport_dimensions**, **-c**[**hemistry_dimensions**], or **-t**[**ransport_dimensions**].

[**X**] [**Y**] [**Z**]--The coordinate directions in which transport occurs. For example, "**Z**" represents 1-dimensional transport in the *Z* direction, "**YZ**" represents 2-dimensional transport in the *Y* and *Z* directions, and "**XYZ**" represents fully 3-dimensional transport. Warning: boundary conditions must be consistent with the defined 1- or 2-dimensional transport or erroneous chemical results will be produced.

Line 8: **-print_orientation** (**XY** or **XZ**)

-print_orientation--The *prefix***.O.**xxx output files contain spatial data printed as a series of planes. Two orientations for the planes are possible, *XY* planes or *XZ* planes. Optionally, **print_orientation** or **-p**[**rint_orientation**].

(**XY** or **XZ**)--Only two orientation options are allowed, either **XY**, indicating *XY* planes are printed or **XZ** indicating *XZ* planes are printed.

Notes

Grid spacing for all three coordinate directions must be defined for all simulations. Each coordinate must be defined with **-uniform** or **-nonuniform**, which are mutually exclusive for each coordinate. The mesh may be refined by adding additional nodes with the identifiers **-overlay_uniform** and **-overlay_nonuniform**. All of the nodes defined are merged and nodes within the distance defined by **-snap** are replaced by a single node.

The identifier **-chemistry_dimensions** is used to save computation time for 1 and 2 dimensional problems. The minimum number of cells (and nodes) in any coordinate direction is two. Flow and transport calculations always involve the entire domain and are thus 3 dimensional. However, if the calculation represents a true 1 or 2 dimensional flow system and boundary conditions are appropriate for 1 or 2 dimensional transport, the symmetry of the transport simulation can be used to reduce the number of geochemical calculations. The geochemical calculations can be performed on a single line or single plane of cells and the results copied to the other lines of

cells or the other plane of cells. If the **-chemistry_dimensions** identifier is not included, the *X*, *Y*, and *Z* directions are active and geochemical calculations are performed for all active cells. For 1- and 2-dimensional geochemical calculations, the number of cells (or nodes) in the inactive coordinate directions must be exactly two.

The use of **-chemistry_dimensions** can lead to erroneous results if the initial and boundary conditions do not allow a truly 1- or 2-dimensional transport simulation. For example, a free-surface calculation could not have zero flow in the *Z* direction. Also, 1- or 2-dimensional flow is not sufficient for 1- or 2-dimensional transport. For example, for steady 1-dimensional flow in the *X* direction, there are four cells in each *YZ* plane. If a contaminant is introduced into only one cell in a *YZ* plane, concentrations are not equal in each of the cells of the *YZ* plane and the transport system is not 1 dimensional; **-chemistry_dimensions** assumes all concentrations perpendicular to the 1-dimensional flow are the same. Thus, boundary and initial conditions must be chosen carefully to ensure 1- or 2-dimensional transport. The program does not check for conceptual errors in the boundary conditions. Empirically, a 1- or 2-dimensional simulation can be tested by removing the **-chemistry_dimensions** identifier and checking that the results are the same as the 1- or 2-dimensional simulation.

HEAD_IC

This keyword is used to define the initial head conditions in the domain. This keyword data block is mandatory for all simulations.

Example 1

```
Line 0:  HEAD_IC
Line 1:        -zone      0     0     0     10    10    10
Line 2:              -head      150
Line 1a:       -zone      5     5     5     10    10    10
Line 2a:             -head      200
```

Explanation 1

Line 0: **HEAD_IC**

HEAD_IC is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone for initial conditions is defined. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone.

Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-head** *property*

-head--Initial heads are defined for the zone. Optionally, **head** or **-h**[**ead**].

property--Heads (L) for the zone may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$

along the given coordinate direction; for locations outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 2 illustrates method 1. Units, L, are defined by **-head** identifier in the **UNITS** data block.

Notes 1

HEAD_IC defines the initial head conditions and, as such, the data block must be present in the data for the first simulation period, that is, the data block must occur before the first **END** keyword in the flow and transport data file. Initial heads are mandatory for all active cells in the domain. Multiple zones may be used within the **HEAD_IC** data block to define the initial conditions for the entire domain. Heads for a single cell may be defined multiple times as part of different zone definitions. The initial head used in the flow or reaction-transport calculations for that cell is the last head defined for it in the flow and transport data file.

Example 2

```
Line 0:  HEAD_IC
Line 1:        -water_table by_cell
Line 2:              150.0 150.5 151.0 151.5 20*152.0
```

Explanation 2

Line 0: **HEAD_IC**

HEAD_IC is the keyword for the data block, no other data are included on this line.

Line 1: **-water_table** *property*

-water_table--Hydrostatic heads are defined by an array of heads, one for each node in the *XY* plane. Optionally, **water_table** or **-w**[**ater_table**].

property--Heads, units L, for the entire *XY* plane may be entered with either of two methods for defining a spatially distributed property: (1) **by_cell** followed by a value for each cell in the *XY* plane, or (2) **file** followed by a file name, a value for each cell in the *XY* plane is read from the file. Line 2 illustrates method 1. Units, L, are defined by **-head** identifier in the **UNITS** data block.

Notes 2

HEAD_IC defines the initial head conditions and, as such, the data block must be present in the data for the first simulation period, that is, the data block must occur before the first **END** keyword in the flow and transport data file. Initial heads are mandatory for all active cells in the domain. The water table heads should only be defined once for the entire *XY* plane. The heads are distributed vertically to all node layers in the model.

LEAKY_BC

This keyword is used to define leaky boundary conditions. For flow-only simulations, only the parameters related to flow for each cell are required. For reaction-transport calculations, the index numbers of an associated solutions are also required. This keyword data block is not needed if no leaky boundary conditions are included in the simulation.

Example

```
Line 0:  LEAKY_BC
Line 1:        -zone      0     10    10    10    10    10
Line 2:              -head            100
Line 3:              -thickness       X     100   0     50    10
Line 4:              -hydraulic_conductivity    file  leaky_k.fil
Line 5:              -associated_solution  5
Line 6:              -face            X
```

Explanation

Line 0: **LEAKY_BC**

LEAKY_BC is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone, which may be a single point (including 1 cell) or a 1- or 2-dimensional zone, for leaky boundary-condition definition. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone. Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-head** *property*

-head--Head on the distal side of the leaky boundary. Optionally, **head** or **-he**[**ad**].

property--The head (L) on the other side of the leaky boundary may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each

cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for locations outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 2 illustrates method 1. Units, L, are defined by the **-head** identifier in the **UNITS** data block.

Line 3: **-thickness** *property*

-thickness--Thickness of the leaky boundary. Optionally, **thickness** or **-t**[**hickness**].

property--The thickness of the leaky boundary (L) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for locations outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 3 illustrates method 4. Units, L, are defined by the **-leaky_thickness** identifier in the **UNITS** data block.

Line 4: **-hydraulic_conductivity** *property*

-hydraulic_conductivity--Hydraulic conductivity in the leaky boundary. Optionally, **hydraulic_conductivity**, **-hy**[**draulic_conductivity**], **k**, or **-k**.

property--The hydraulic conductivity of the boundary layer (L/T) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each

cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for locations outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 4 illustrates method 3. Units, L/T, are defined by the **-leaky_hydraulic_conductivity** identifier in the **UNITS** data block.

Line 5: **-associated_solution** *property*

-associated_solution--Solution index numbers associated with the cells with leaky boundary conditions. Optionally, **associated_solution**, **-a**[**ssociated_solution**], **solution**, or **-s**[**olution**].

property--Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Solutions or mixtures of solutions may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 5 illustrates method 1.

Line 6: **-face** (**X**, **Y**, or **Z**)

-face--If the zone is 0- or 1-dimensional, the coordinate direction for the leakage must be specified.

Optionally, **face** or **-fa**[**ce**].

X, **Y**, or **Z**--The coordinate direction of the leakage.

Notes

The zones for all boundary conditions must be 0-, 1-, or 2-dimensional. For leaky boundary conditions with 2-dimensional zones, the coordinate direction for the leakage can be determined from the zone definition and the **-face** identifier is not necessary. For 0- and 1-dimensional zones the coordinate direction for the leakage is not known from the zone definition, and the **-face** identifier must be used to identify the coordinate direction of the leakage.

Multiple zones may be used within the **FLUX_BC**, **LEAKY_BC**, and **SPECIFIED_VALUE_BC** data blocks to define boundary conditions within the domain. Different boundary conditions for a single cell (node) may be defined multiple times as part of different zone definitions and different keyword data blocks. The boundary condition that is used for a cell is the last zone definition that defines a boundary condition for that cell.

MEDIA

This keyword is used to define media properties, including hydraulic conductivities, porosity, specific storage, and dispersivities. These spatial properties are applied to elements, not cells. This keyword data block is mandatory in the data for the first simulation period for both flow and reaction-transport simulations.

Example

```
Line 0:  MEDIA
Line 1:        -zone       0     0     0     10    10    10
Line 2:             -Kx          file  kx.fil
Line 3:             -Ky          file  ky.fil
Line 4:             -Kz          file  kz.fil
Line 5:             -porosity    X     0.2   0     0.3   10
Line 6:             -specific_storage     0
Line 7:             -trans_dispersivity   by_element
Line 7a:                 10*2.0 10*4.0
Line 8:             -long_dispersivity    2
Line 1a:       -zone       4     0     0     5     0     0
Line 9:             -active     0
```

Explanation

Line 0: **MEDIA**

MEDIA is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone for which media properties are defined. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone.

Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-Kx** *property*

-Kx--Hydraulic conductivity in the *X* direction. Optionally, **Kx**, **Kxx**, or **-Kx**[**x**].

property--The hydraulic conductivity in the *X* direction (L/T) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2)

by_element followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 2 illustrates method 3. Units, L/T, are defined by the **-hydraulic_conductivity** identifier in the **UNITS** data block.

Line 3: **-Ky** *property*

-Ky--Hydraulic conductivity in the *Y* direction. Optionally, **Ky**, **Kyy**, or **-Ky**[**y**].

property--The hydraulic conductivity in the *Y* direction (L/T) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element** followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 3 illustrates method 3. Units, L/T, are defined by the **-hydraulic_conductivity** identifier in the **UNITS** data block.

Line 4: **-Kz** *property*

-Kz--Hydraulic conductivity in the *Z* direction. Optionally, **Kz**, **Kzz**, or **-Kz**[**z**].

property--The hydraulic conductivity in the *Z* direction (L/T) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2)

by_element followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 4 illustrates method 3. Units, L/T, are defined by the **-hydraulic_conductivity** identifier in the **UNITS** data block.

Line 5: **-porosity** *property*

-porosity--Porosity of the medium. Optionally, **porosity** or **-p**[**orosity**].

property--The porosity (unitless) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element** followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 5 illustrates method 4.

Line 6: **-specific_storage** *property*

-specific_storage--Specific storage for the zone. Optionally, **specific_storage**, **storage**, **-s**[**pecific_storage**], or **-s**[**torage**].

property--The specific storage for the zone (1/L) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element**

followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 6 illustrates method 1. Units, 1/L, are defined by the **-specific_storage** identifier in the **UNITS** data block.

Line 7: **-trans_dispersivity** *property*

-trans_dispersivity--Dispersivity transverse to the direction of the flow velocity vector. Optionally, **trans_dispersivity**, **transverse_dispersivity**, **trans**, **-t**[**rans_dispersivity**], or **-t**[**ransverse_dispersivity**].

property--The dispersivity transverse to the direction of the flow-velocity vector (L) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element** followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 7 illustrates method 2 (note use of repeat factors to enter 10 values of 2.0 and 10 values of 4.0). Units, L, are defined by the **-dispersivity** identifier in the **UNITS** data block.

Line 8: **-long_dispersivity** *property*

-long_dispersivity--Dispersivity in the direction of the flow velocity vector. Optionally, **long_dispersivity**, **longitudinal_dispersivity**, **long**, **-l**[**ong_dispersivity**], or **-l**[**ongitudinal_dispersivity**].

property--Dispersivity in the direction of the flow-velocity vector (L) may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element** followed by a value for each element centroid in the zone, (3) **file** followed by a file name, a value for each element centroid in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each element centroid in the zone; interpolation is proportional to the fractional distance of the element centroid between $distance_1$ and $distance_2$ along the given coordinate direction; for centroids outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Line 8 illustrates method 1. Units, L, are defined by the **-dispersivity** identifier in the **UNITS** data block.

Line 9: **-active** *property*

-active--Definition of active and inactive elements. Optionally, **active** or **-a**[**ctive**].

property--Active and inactive elements are defined with values of 1 (active) and 0 (inactive). The active and inactive elements may be entered with any of three methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_element** followed by a value for each element in the zone, or (3) **file** followed by a file name, a value for each element in the zone is read from the file. Line 9 illustrates method 1.

Notes

Media properties are defined by element, not by cell. The number of elements is one fewer in each coordinate direction than the number of nodes. The number of elements listed using property input methods 2 and 3 must match exactly the number of elements with centroids that are within the zone definition.

If a free surface is simulated (**FREE_SURFACE_BC** data block), the specific storage is reset to zero regardless of the value entered in the **MEDIA** data block.

Inactive cells may not be defined for 1-dimensional simulations (**-chemistry_dimensions** in **GRID** data block).

Multiple zones may be used within the **MEDIA** data block to media properties within the domain. Different media properties for a single element may be defined multiple times as part of different, overlapping zone definitions. An individual media property that is used for an element is the last zone definition that defines that media property for that element.

PRINT_FREQUENCY

This keyword is used to select which results are written to output files and the frequency at which the results are written. The data block and all identifiers are optional. By default, printing controlled by all identifiers that are indicated by a zero or "false" value in the example will not be written; printing controlled by all identifiers that are indicated by a nonzero value in the example will be written to files at the end of each simulation period.

Example

```
Line 0:  PRINT_FREQUENCY
Line 1:       -bc_flow_rates              0     days
Line 2:       -boundary_conditions        False
Line 3:       -components                 0     days
Line 4:       -conductances               0     yr
Line 5:       -flow_balance               1     step
Line 6:       -force_chemistry_print      0     step
Line 7:       -HDF_chemistry              1     yr
Line 8:       -HDF_heads                  1     yr
Line 9:       -HDF_velocities             1     yr
Line 10:      -heads                      1     yr
Line 11:      -progress_statistics        1     step
Line 12:      -save_final_heads           False
Line 13:      -velocities                 0     yr
Line 14:      -wells                      2     yr
Line 15:      -xyz_chemistry              0     days
Line 16:      -xyz_components             0     yr
Line 17:      -xyz_heads                  0     yr
Line 18:      -xyz_velocities             0     yr
Line 19:      -xyz_wells                  0     step
```

Explanation

Line 0: **PRINT_FREQUENCY**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-bc_flow_rates** *frequency*, (*time unit* or **step**)

-bc_flow_rates--Controls writing of the flow rates for all boundary-condition cells to the file *prefix***.O.bcf**. Until a simulation period when **-bc_flow_rates** is defined, writing will occur at the end of each simulation period. Data in the file are formatted in 2D planes as defined by

-print_orientation in the **GRID** data block of the flow and transport data file. Optionally, **bc_flow_rates** or **-bc_**[**flow_rates**].

frequency--Frequency at which flow rates are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O**.**bcf**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 2: **-boundary_conditions** [***True*** *or* ***False***]

-boundary_conditions--For each stress period, heads, component concentrations, fluxes, and other information related to boundary conditions is written to the file *prefix***.O.probdef**. Optionally, **boundary_conditions**, **boundary**, **bc**, **-b**[**c**], or **-b**[**oundary_conditions**].

True *or* ***False***--**True** writes boundary-condition information to the file *prefix***.O.probdef**; **false** excludes write. If the identifier is not included in the input file, the default initialization is **false**. Optionally, **t**[**rue**] or **f**[**alse**], case independent. By default, no data will be written.

Line 3: **-components** *frequency*, (*time unit* or **step**)

-components--Controls writing of total element (component) data for each cell to the file *prefix*.**O**.**comps**. Until a simulation period when **-components** is defined, writing to the file will not occur. Data in the file are formatted in 2D planes as defined by **-print_orientation** in the **GRID** data block of the flow and transport data file. Writing of initial condition component concentrations is controlled by **-components** identifier in the **PRINT_INITIAL** data block. Optionally, **component**, **components**, or **-com**[**ponents**].

frequency--Frequency at which component concentration data are written to files. Frequency may be either an interval of time or a number of time steps between writing results to the file. If

frequency is zero, no transient data will be written to the file *prefix*.**O**.**comps**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 4: **-conductances** *frequency*, (*time unit* or **step**)

-conductances--Controls writing of transient fluid and dispersive conductances for each cell face to the file *prefix***.O.kd**. Until a simulation period when **-conductances** is defined, writing to the file will not occur. Data in the file are formatted in 2D planes as defined by **-print_orientation** in the **GRID** data block of the flow and transport data file. Optionally, **conductances, conductance,** or **-cond**[**uctances**].

frequency--Frequency at which transient conductances are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O**.**kd**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 5: **-flow_balance** *frequency*, (*time unit* or **step**)

-flow_balance--Controls writing of the flow balance information for the domain to the file *prefix***.O.bal**. Until a simulation period when **-flow_balance** is defined, writing will occur at the end of each simulation period. Optionally, **flow_balance** or **-f**[**low_balance**].

frequency--Frequency at which flow-balance data are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no

transient data will be written to the file *prefix*.**O**.**bal**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 6: **-force_chemistry_print** *frequency*, (*time unit* or **step**)

-force_chemistry_print--Controls writing of detailed chemical descriptions of the composition of the solution and all reactants for each cell to the file *prefix***.O.chem**. Until a simulation period when **-force_chemistry_print** is defined, writing to the file will not occur. *Warning*: this file could be huge because every selected print interval will produce a long description of each cell. Writing of this information may be useful for debugging, for small problems, or if the cells for which writing results are restricted by cell selections made in the **PRINT_LOCATIONS** data block of the flow and transport data file. Data written to the file can be restricted by options within the **PRINT** data block of the chemical data file. Optionally, **force_chemistry**, **force_chemistry_print**, or **-fo**[**rce_chemistry_print**].

frequency--Frequency at which chemical data are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O.chem**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 7: **-HDF_chemistry** *frequency*, (*time unit* or **step**)

-HDF_chemistry--Controls writing of chemistry data to the file *prefix***.h5**. Until a simulation period when **-HDF_chemistry** is defined, writing to the file will occur at the end of each simulation period. The file is a binary Hierarchical Data Format file that can be read by the ModelViewer program or by using HDF version 5 utilities. Data are stored in natural node order. Chemistry data to be written to the file *prefix***.h5** (and *prefix***.xyz.chem**) are defined in the **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file. Writing of initial condition chemistry data is controlled by **-HDF_chemistry** identifier in the **PRINT_INITIAL** data block. Optionally, **hdf_chemistry**, **hdf_concentration**, **hdf_concentrations**, **-hdf_c**[**oncentrations**], or **-hdf_c**[**hemistry**], case independent.

frequency--Frequency at which chemistry data are written to the HDF file. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no chemistry data will be written to the file *prefix*.**h5**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be “seconds”, “minutes”, “hours”, “days”, or “years” or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 8: **-HDF_heads** *frequency*, (*time unit* or **step**)

-HDF_heads--Controls writing of heads to the file *prefix***.h5**. Until a simulation period when **-HDF_heads** is defined, writing will occur at the end of each simulation period. The file is a binary Hierarchical Data Format file that can be read by the ModelViewer program or by using HDF version 5 utilities. Writing of initial condition heads is controlled by the **-hdf_heads** identifier in the **PRINT_INITIAL** data block. Data are stored in natural node order. Optionally, **hdf_head**, **hdf_heads**, or **-hdf_h**[**eads**], case independent.

frequency--Frequency at which head data are written to the HDF file. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no head data will be written to the file *prefix*.**h5**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 9: **-HDF_velocities** *frequency*, (*time unit* or **step**)

-HDF_velocities--Controls writing of X, Y, and Z velocities to the file *prefix***.h5**. Until a simulation period when **-HDF_velocities** is defined, writing to the file will occur at the end of each simulation period. The file is a binary Hierarchical Data Format file that can be read by the ModelViewer program or by using HDF version 5 utilities. Data are stored in natural node order. If **-HDF_velocities** is not defined, writing will occur at the end of each simulation period. Optionally, **hdf_velocity**, **hdf_velocities**, **-hdf_v**[**elocities**], or **-hdf_v**[**elocity**], case independent.

frequency--Frequency at which head data are written to the HDF file. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no head data will be written to the file *prefix*.**h5**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 10: **-heads** *frequency*, (*time unit* or **step**)

-heads--Controls writing of heads to the file *prefix***.O.head**. Until a simulation period when **-heads** is defined, writing to the file will occur at the end of each simulation period. Data in the file are formatted in 2D planes as defined by **-print_orientation** in the **GRID** data block of the flow and transport data file. Writing of initial condition heads is controlled by **-heads** identifier in the **PRINT_INITIAL** data block. Optionally, **head**, **heads**, or **-h**[**eads**].

frequency--Frequency at which head data are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O.head**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 11: **-progress_statistics** *frequency*, (*time unit* or **step**)

-progress_statistics--Controls writing of solver statistics, including solution-method information, number of iterations, and maximum changes in head to the file *prefix***.log** and to the screen. Until a simulation period when **-progress_statistics** is defined, writing to the file will occur at the end of each simulation period. Optionally, **progress_statistics**, **-pr**[**ogress_statistics**], **solver_statistics** or **-solv**[**er_statistics**].

frequency--Frequency at which data are solver statistics are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no data on solver iterations will be written to the file *prefix*.**log**. By default, printing will occur at the end of each simulation period.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every

frequency time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 12: **-save_final_heads** [***True*** *or* ***False***]

-save_final_heads--At the end of the run, heads are written to the file *prefix***.head.dat** in a form that can be read for initial conditions in subsequent runs. The file *prefix***.head.dat** can be used for initial head conditions in subsequent runs by using a zone that includes the entire domain and "**-head file** *prefix***.head.dat**" in **HEAD_IC** data block. Optionally, **save_head**, **save_heads**, **save_final_heads**, **-sa**[**ve_heads**], or **-sa**[**ve_final_heads**].

True *or* ***False***--**True** writes heads at the end of the run to the file *prefix***.head.dat**; **false** excludes write. Default is **true** if neither ***True*** nor ***False*** is entered; initial setting is **false**. Optionally, **t**[**rue**] or **f**[**alse**], case independent. By default, no data will be written.

Line 13: **-velocities** *frequency*, (*time unit* or **step**)

-velocities--Controls writing of interstitial velocities at cell boundaries and velocities at nodes to the file *prefix***.O.vel**. Until a simulation period when **-velocities** is defined, writing to the file will occur at the end of each simulation period. Data in the file are formatted in 2D planes as defined by **-print_orientation** in the **GRID** data block of the flow and transport data file. Optionally, **velocities**, **velocity -v**[**elocities**], or **-v**[**elocity**].

frequency--Frequency at which interstitial velocities at cell boundaries are written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O.vel**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 14: **-wells** *frequency*, (*time unit* or **step**)

-wells--Controls writing of transient well information, including fluid and solute flow rates, cumulative fluid and solute flow amounts, and solute concentrations, to the file *prefix*.**O.wel**. Until a simulation period when **-wells** is defined, writing to the file will occur at the end of each simulation period. Data are written in the order of the well sequence numbers. Writing of initial condition well information is controlled by the **-wells** identifier in the **PRINT_INITIAL** data block. Optionally, **wells** or **-w**[**ells**].

frequency--Frequency at which transient well information is written. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.**O.wel**.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 15: **-xyz_chemistry** *frequency*, (*time unit* or **step**)

-xyz_chemistry--Controls writing of selected chemical data to the file *prefix*.**xyz.chem**. Until a simulation period when **-xyz_chemistry** is defined, writing to the file will not occur. X, Y, and Z locations, time, and a flag to indicate whether the cell is dry, followed by selected data are written to the file, one line for each active cell. Cells for which results are to be written can be selected with the **PRINT_LOCATIONS** data block of the flow and transport data file. The **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file are used to define the data that are written to the file *prefix*.**xyz.chem**. Writing of initial concentrations of components is controlled by the **-xyz_chemistry** identifier in the **PRINT_INITIAL** data block. Optionally, **concentrations**, **selected_output**, **selected_outputs**, **xyz_chemistry**, **-c**[**oncentrations**], **-se**[**lected_outputs**], or **-xyz_ch**[**emistry**].

frequency--Frequency at which transient concentration data are written to files. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix***.xyz.chem**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 16: **-xyz_components** *frequency*, (*time unit* or **step**)

-xyz_components--Controls writing of element (component) concentrations to the file *prefix***.xyz.comps**. Until a simulation period when **-xyz_components** is defined, writing to the file will not occur. If **-xyz_components** is never defined in a run, writing to the file will not occur. X, Y, and Z locations, time, and a flag to indicate whether the cell is dry, followed by component concentrations are written to the file, one line for each active cell. Data for cells are written in the sequence of increasing X, then Y, then Z. Writing of initial concentrations of components is controlled by the **-xyz_components** identifier in the **PRINT_INITIAL** data block. Optionally, **xyz_component**, **xyz_components**, or **-xyz_c**[**omponents**].

frequency--Frequency at which head data are written to the file *prefix***.xyz.comps**. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix*.xyz**.comps**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 17: **-xyz_heads** *frequency*, (*time unit* or **step**)

-xyz_heads--Controls writing of heads to the file *prefix***.xyz.head**. Until a simulation period when **-xyz_heads** is defined, writing to the file will not occur. X, Y, and Z locations, time, and a flag to indicate whether the cell is dry, followed by head are written to the file, one line for each active cell. Data for cells are written in the sequence of increasing X, then Y, then Z. Writing of initial condition heads is controlled by the **-xyz_heads** identifier in the **PRINT_INITIAL** data block. Optionally, **xyz_head**, **map_head**, **-xyz_h**[**ead**], or **-map_h**[**ead**].

frequency--Frequency at which head data are written to the file *prefix***.xyz.head**. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix***.xyz.head**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 18: **-xyz_velocities** *frequency*, (*time unit* or **step**)

-xyz_velocities--Controls writing of velocities at cell nodes to the file *prefix***.xyz.vel**. Until a simulation period when **-xyz_velocities** is defined, writing to the file will not occur. X, Y, and Z locations, time, and a flag to indicate whether the cell is dry, followed by X, Y, and Z velocities are written to the file, one line for each active cell. Data for cells are written in the sequence of increasing X, then Y, then Z. Optionally, **xyz_velocity**, **map_velocity**, **-xyz_v**[**elocity**], or **-map_v**[**elocity**].

frequency--Frequency at which velocities at cell nodes are written to the file *prefix***.xyz.vel**. Frequency may be either an interval of time or a number of time steps between writing results to the file.

If *frequency* is zero, no transient data will be written to the file *prefix***.xyz.vel**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Line 19: **-xyz_wells** *frequency*, (*time unit* or **step**)

-xyz_wells--Controls writing of a time-series of concentrations for each well to the file *prefix***.xyz.wel**. Until a simulation period when **-xyz_wells** is defined, writing to the file will not occur. The file includes a line for each well at each selected time. Optionally, **xyz_well**, **xyz_wells**, **well_time_series**, **-xyz_w**[**ells**], **-well_**[**time_series**], **wells_time_series**, or **-wells_**[**time_series**].

frequency--Frequency at which concentrations are written to the file *prefix***.xyz.wel**. Frequency may be either an interval of time or a number of time steps between writing results to the file. If *frequency* is zero, no transient data will be written to the file *prefix***.xyz.wel**. By default, no data will be written.

time unit or **step**--*Time unit* may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Alternatively, if **step** is entered, results are written every *frequency* time steps in the calculation. Default units are defined by **-time** identifier in **UNITS** data block.

Notes

User-specified units for the various print frequencies need not be the same. One **PRINT_FREQUENCY** data block may contain multiple time units and (or) the **step** unit. If time units are used for a print frequency, and the simulation time does not fall on an even multiple of the print frequency, then when the simulation time is within one time step of an even multiple of the print frequency, the time step will be decreased to produce a simulation

time for which printing will occur. The next step will revert to the original time step, unless a smaller time step is needed to reach another print-frequency criterion.

A value of 0 (zero) for *frequency* eliminates writing of the specified data. All print frequencies that are not defined, except for **-components**, **-conductances**, **-force_chemistry_print**, **-xyz_components**, **-xyz_heads**, **-xyz_velocities**, **-wells**, and **-xyz_wells**, are set such that writing will occur at the end of each simulation period. If the print frequency for **-components**, **-conductances**, **-force_chemistry_print**, **-xyz_components**, **-xyz_heads**, **-xyz_velocities**, **-wells**, or **-xyz_wells** is not defined, no writing will occur to the file that the option controls. Once set by the keyword data block, **PRINT_FREQUENCY**, options will remain in effect until the end of the run or until changed in another **PRINT_FREQUENCY** data block.

The **xyz_wells** identifier can only be defined at the first simulation period; it can not be redefined to another frequency in a subsequent simulation period. All other print frequencies in this data block may be redefined for each simulation period if desired. Writing to the files is determined by number of time steps or total time since the beginning of the run, not from the beginning of the simulation period. For example, if a print frequency is specified to be every 2 steps in the first simulation period and every 3 steps in the second, and both simulation periods have 5 time steps, the printing will occur at time steps 2 (evenly divisible by 2), 4, and 5 (last time step of simulation period is printed) in the first simulation period, and at time steps 6 (the first time step of the second simulation period, but evenly divisible by 3), 9, and 10 (last time step of simulation period) in the second simulation period.

When steady-state flow is simulated (**STEADY_FLOW true**), nonzero print frequencies for the identifiers **-head** (*prefix***.O.head**), **-flow_balance** (*prefix***.O.bal**), **-velocity** (*prefix***.O.vel)**, **-conductances** (*prefix***.O.kd**), **-hdf_heads** (*prefix***.h5**), **-hdf_velocities** (*prefix***.hdf.vel**), **-xyz_heads** (*prefix***.xyz.head**), and **-xyz_velocities** (*prefix***.xyz.vel**) are treated specially. During the iterations to achieve steady-state flow, heads are written to the *prefix***.O.head** file for every steady-state iteration; no data are written to this file during the transient part of the calculation. During the iterations to achieve steady-state flow, balances are written to the *prefix***.O.bal** file for every steady-state iteration; the print frequency defined by **-flow_balance** is used during the transient part of the simulation. Velocities are written to the *prefix***.O.vel**, *prefix***.h5**, and *prefix***.xyz.vel** files, heads are written to the

*prefix***.h5** and *prefix***.xyz.head**, and conductance factors are written to the file *prefix***.O.kd** only once after steady-state flow has been calculated, provided the *frequency* for the corresponding identifier is nonzero for any stress period within the flow and transport data file. Printing of steady-flow heads and velocities can also be requested in the **PRINT_INITIAL** data block.

The *prefix* used for all of the file names is defined interactively at the time PHAST is invoked. The files names containing "**.O.**" are data formatted to be printed or viewed on a screen. The orientation of the printout of the spatially distributed properties in the "**.O.**" files, either *XY* or *XZ* planes, is controlled by the **-print_orientation** identifier in the **GRID** data block. The files *prefix***.xyz.chem**, *prefix***.xyz.comps**, *prefix***.xyz.head**, *prefix***.xyz.vel**, and *prefix***.xyz.wel** are written in a form to facilitate importing into spreadsheets and writing post-processing programs for graphical display.

PRINT_INITIAL

This keyword is used to print the static flow and transport data to various output files. The options are useful to ensure that media properties, and initial and boundary conditions have been defined correctly. Default values for the options are given in the example.

Example

```
Line 0:  PRINT_INITIAL
Line 1:        -boundary_conditions       false
Line 2:        -components                false
Line 3:        -conductances              false
Line 4:        -echo_input                true
Line 5:        -fluid_properties          true
Line 6:        -force_chemistry_print     false
Line 7:        -HDF_chemistry             true
Line 8:        -HDF_heads                 true
Line 9:        -HDF_steady_flow_velocities true
Line 10:       -heads                     true
Line 11:       -media_properties          false
Line 12:       -solution_method           true
Line 13:       -steady_flow_velocities    false
Line 14:       -wells                     true
Line 15:       -xyz_chemistry             false
Line 16:       -xyz_components            false
Line 17:       -xyz_heads                 false
Line 18:       -xyz_steady_flow_velocities false
Line 19:       -xyz_wells                 false
```

Explanation

Line 0: **PRINT_INITIAL**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-boundary_conditions** [***True*** *or* ***False***]

-boundary_conditions--Print boundary condition information to the file *prefix***.O.probdef**, including data for specified head, specified flux, and leaky boundary conditions and the concentrations of all components in solutions related to boundary conditions. Optionally, **boundary_conditions**, **boundary**, **bc**, **-b**[**oundary_conditions**], or **-b**[**c**].

True** or **False--**True** prints static boundary-condition data to the file *prefix***.O.probdef**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 2: **-components** [***True** or **False***]

-components--Print initial component concentrations and indices and mixing fractions for solutions, equilibrium phases, exchangers, surfaces, gas phases, solid solutions, and kinetic reactions that define initial conditions for the simulation to the file *prefix***.O.comps**. Optionally, **component**, **components**, or **-c**[**omponents**].

True** or **False--**True** prints initial component concentrations and indices and mixing fractions to the file *prefix***.O.comps**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 3: **-conductances** [***True** or **False***]

-conductances--Print the static fluid conductance factors to the file *prefix***.O.kd**. Optionally, **conductance**, **conductances**, or **-con**[**ductances**].

True** or **False--**True** prints fluid conductance factors to the file *prefix***.O.kd**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 4: **-echo_input** [***True** or **False***]

-echo_input--This identifier is used to determine whether lines from the flow and transport data file are written to the file *prefix***.log** as they are processed. The option take effect as soon as it is encountered in the flow and transport data file. (Writing lines from the chemical data file to the file *prefix***.log** is controlled by **-echo_input** in the **PRINT** data block of the chemical data file.) Optionally, **echo_input** or **-e**[**cho_input**].

True** or **False--**True** writes lines from the flow and transport data file to the file *prefix***.log** as they are processed; **false** excludes write. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 5: **-fluid_properties** [***True** or **False***]

-fluid_properties--Print input data for all the fluid properties to the file *prefix***.O.probdef**, including compressibility, molecular diffusivity, viscosity, and density. Optionally, **fluid_properties**, **fluid**, or **-f**[**luid_properties**].

True *or* ***False***--**True** prints input fluid properties to the file *prefix***.O.probdef**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 6: **-force_chemistry_print** *frequency*, (*time unit* or **step**)

-force_chemistry_print--Controls writing of detailed chemical descriptions of the composition of the solution and all reactants for each cell to the file *prefix***.O.chem**. *Warning*: this file could be huge because it will produce a long description of the chemistry of each cell. Writing of this information may be useful for debugging, for small problems, or if the cells for which writing results are restricted by cell selections made in the **PRINT_LOCATIONS** data block of the flow and transport data file. Data written to the file can be restricted by options within the **PRINT** data block of the chemical data file. Optionally, **force_chemistry**, **force_chemistry_print**, or **-fo**[**rce_chemistry_print**].

True *or* ***False***--**True** prints detailed chemical description of each cell to the file *prefix***.O.chem**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 7: **-HDF_chemistry** [***True*** *or* ***False***]

-HDF_chemistry--Print chemistry data that is calculated from initial conditions to the file *prefix***.h5**. Data to be printed are defined in **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file. Optionally, **hdf_chemistry**, **hdf_concentration**, **hdf_concentrations**, **-hdf_c**[**hemistry**], or **-hdf_c**[**oncentrations**], case independent.

True *or* ***False***--**True** prints initial chemistry data to the file *prefix***.h5**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 8: **-HDF_heads** [***True*** *or* ***False***]

-HDF_heads--Print initial heads to the file *prefix***.h5**. Optionally, **hdf_head**, **hdf_heads**, or **-hdf_h**[**eads**], case independent.

True *or* ***False***--**True** prints initial heads to the file *prefix***.h5**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 9: **-HDF_steady_flow_velocities** [***True*** *or* ***False***]

-HDF_steady_flow_velocities--Print steady-flow velocities to the file *prefix***.h5**. This option has meaning only if steady flow is specified in the **STEADY_FLOW** data block. Optionally, **hdf_steady_flow_velocity**, **hdf_steady_flow_velocities**, **-hdf_s**[**teady_flow_velocity**], **-hdf_s**[**teady_flow_velocities**], **hdf_ss_velocity**, **hdf_ss_velocities**, **-hdf_s**[**s_velocity**], or **-hdf_s**[**s_velocities**], case independent.

True *or* ***False***--**True** prints steady-flow velocities to the file *prefix***.h5**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 10: **-heads** [***True*** *or* ***False***]

-heads--Print initial heads to the file *prefix***.O.head**. Optionally, **head**, **heads**, or **-h**[**eads**].

True *or* ***False***--**True** prints initial heads to the file *prefix***.O.head**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 11: **-media_properties** [***True*** *or* ***False***]

-media_properties--Print input data for all the media properties, porous media zones, including element zone definitions, porosities, hydraulic conductivities, dispersivities, and other information related to media properties. Optionally, **media_properties**, **media**, **medium**, **-m**[**edia_properties**], or **-m**[**edium**].

True *or* ***False***--**True** prints input media properties, including hydraulic conductivity, porosity, dispersivity, and specific storage, to the file *prefix***.O.probdef** and also prints fluid and dispersive conductance factors to the file *prefix***.O.kd**; **false** excludes printing these data to the files. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 12: **-solution_method** [***True*** *or* ***False***]

-solution_method--Print input data for the solution method for flow and transport equations to the file *prefix***.O.probdef**, including type of solver used and any solver parameters. Optionally, **solution_method**, **method**, **-s**[**olution_method**], or **-met**[**hod**].

True *or* ***False***--**True** prints data for the solution method to the file *prefix***.O.probdef**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 13: **-steady_flow_velocities** [***True*** *or* ***False***]

-steady_flow_velocities--Print steady-flow velocities to the file *prefix***.O.vel**. This option has meaning only if steady flow is specified in the **STEADY_FLOW** data block. Optionally, **steady_flow_velocity**, **steady_flow_velocities**, **-st**[**eady_flow_velocity**], **-st**[**eady_flow_velocities**], **ss_velocity**, **ss_velocities**, **-ss**[**_velocity**], or **-ss**[**_velocities**], case independent.

True *or* ***False***--**True** prints steady-flow velocities to the file *prefix***.O.vel**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 14: **-wells** [***True*** *or* ***False***]

-wells--Print static well information, including location, diameter, screened intervals, and well indices. Optionally, **wells**, or **-w**[**ells**].

True *or* ***False***--**True** prints static well information to the file *prefix***.O.wel**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **true**.

Line 15: **-xyz_chemistry** [***True*** *or* ***False***]

-xyz_chemistry--Print initial chemistry data to the file *prefix***.xyz.chem**. Optionally, **xyz_chemistry** or **-xyz_ch**[**emistry**]. The **SELECTED_OUTPUT** and **USER_PUNCH** data blocks of the chemical data file are used to define the data that are written to the file *prefix***.xyz.chem**.

True *or* ***False***--**True** prints initial component concentrations to the file *prefix***.xyz.head**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 16: **-xyz_components** [***True*** *or* ***False***]

-xyz_components--Print initial component concentrations to the file *prefix***.xyz.comps**. Optionally, **xyz_component**, **xyz_components**, or **-xyz_c**[**omponents**].

True *or* ***False***--**True** prints initial component concentrations to the file *prefix***.xyz.head**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 17: **-xyz_heads** [***True*** *or* ***False***]

-xyz_heads--Print initial heads to the file *prefix***.xyz.head**. Optionally, **xyz_head**, **xyz_heads**, or **-xyz_h**[**eads**], case independent.

True *or* ***False***--**True** prints initial heads to the file *prefix***.xyz.head**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 18: **-xyz_steady_flow_velocities** [***True*** *or* ***False***]

-xyz_steady_flow_velocities--Print velocities from the steady-flow calculation to the file *prefix***.xyz.vel**. This option has meaning only if steady flow is specified in the **STEADY_FLOW** data block. Optionally, **xyz_steady_flow_velocity**, **xyz_steady_flow_velocities**, **-xyz_s**[**teady_flow_velocity**], **-xyz_s**[**teady_flow_velocities**], **xyz_ss_velocity**, **xyz_ss_velocities**, **-xyz_s**[**s_velocity**], or **-xyz_s**[**s_velocities**].

True *or* ***False***--**True** prints initial velocities to the file *prefix***.xyz.vel**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Line 19: **-xyz_wells** [***True*** *or* ***False***]

-xyz_wells--Print initial concentrations at wells to the file *prefix***.xyz.wel**. Optionally, **xyz_well**, **xyz_wells**, or **-xyz_w**[**ells**], case independent.

True *or* ***False***--**True** prints initial concentrations to the file *prefix***.xyz.wel**; **false** excludes print. Optionally, **t**[**rue**] or **f**[**alse**], case independent. Initial setting is **false**.

Notes

By default, settings at the beginning of a run are **true** for **-echo_input**, **-fluid_properties**, **-HDF_chemistry**, **-HDF_heads**, -**HDF_steady_flow_velocities**, **-heads**, **-solution_method**, and **-wells**; settings at the beginning of a run are false for all other identifiers. The **PRINT_INITIAL** options are used only once, prior to any transient calculations. The orientation of the printout of the spatially distributed properties to the *prefix***.O** files, either *XY* or *XZ* planes, is controlled by the **-orientation** identifier in the **GRID** data block. Properties for inactive cells are shown as blanks in the appropriate location in the *prefix***.O** files. The *prefix* used for all of the file names is defined at the time PHAST is invoked.

PRINT_LOCATIONS

This keyword data block is used to limit printing of results to the chemistry output file (*prefix*.**O.chem**) and the file *prefix*.**xyz.chem** to a subset of cells within the model domain. In the absence of this data block, results are printed to the files *prefix*.**O.chem** and *prefix*.**xyz.chem** for all active cells if printing to these files is enabled. The **PRINT**, **SELECTED_OUTPUT** and **USER_PUNCH** data blocks in the chemistry data file define the data to be included in the files *prefix*.**O.chem** and *prefix*.**xyz.chem**. The identifiers **-force_chemistry_print** and **-concentration** in **PRINT_FREQUENCY** specify the time steps for which results will be printed to the files *prefix*.**O.chem** and *prefix*.**xyz.chem**.

Example

```
Line 0:  PRINT_LOCATIONS
Line 1:        -sample X 2
Line 2:        -zone      5     0     0     10    10    10
Line 3:             -print     1
```

Explanation

Line 0: **PRINT_LOCATIONS**

PRINT_LOCATIONS is the keyword for the data block, no other data are included on this line.

Optionally, **PRINT_LOCATION**.

Line 1: **-sample** (**X**, **Y**, or **Z**) *sample frequency*

-sample--Printing to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** is defined for a subgrid. Optionally, **sample**, **sample_grid**, **thin**, **thin_grid**, **-s**[**ample_grid**] or **-t**[**hin_grid**].

X, **Y**, or **Z**--The coordinate direction for which grid sampling for printing to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** is to be defined.

sample frequency--Printing to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** will occur for the first and last nodes in the coordinate direction (provided these nodes are active), and, between the first and last nodes, results will be printed at for node intervals of *sample frequency*. If *sample frequency* is 2, then every other node will be printed. The nodes are selected by working to the interior of the domain from each end of the coordinate direction.

Line 2: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone for which printing is defined. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone. Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 3: **-print** *property*

-print--Specification of print activity to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** for cells within the zone. Optionally, **print** or **-p**[**rint**].

property--A value of 1 indicates results will be printed to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** for cells in the zone; a value of 0 indicates results will not be printed. Values for **print** may be entered with any of three methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell within the zone, or (3) **file** followed by a file name, a value for each cell in the zone is read from the file. Line 2 illustrates method 1.

Notes

By default, printing to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** will occur for all active cells (**MEDIA**, **-active**) for the dimensions for which chemistry is calculated (**GRID**, **-chemistry_dimensions**). Although, the cells to be printed to the files *prefix*.**xyz.chem** and *prefix*.**O.chem** can be changed for any stress-period, normally, the **PRINT_LOCATIONS** data block will normally be included in the definition of the first stress period and not changed in subsequent stress periods. If both **-sample_grid** and **-print** identifiers are defined, the **-print** definitions supersede the **-sample_grid** definitions. If a cell is included in more than one zone, the definition of **-print** in the last definition will apply. Printing will be defined by the accumulation of all the **PRINT_LOCATIONS** data blocks from the beginning of the run through the current stress period. Cells to be printed to the files

prefix.**xyz.chem** and *prefix*.**O.chem** are restricted if **-chemistry_dimensions** in the **GRID** data block is not 3-dimensional and if zones are defined to be inactive (**MEDIA** data block).

RIVER

This keyword is used to define a river boundary condition. Multiple **RIVER** data blocks may be used to define all of the rivers in the domain.

Example

```
Line 0:  RIVER 1 Rubicon River
Line 1:        -default_width        150.
Line 2:        -default_depth        4.0
Line 3:        -default_bed_hydraulic_conductivity   1e-2
Line 4:        -default_bed_thickness                1.0
Line 5:        -default_solution     2
Line 6:        155.0      3633.  # x and y location of point on river
Line 7:             -head                 275.
Line 8:             -width                125.
Line 9:             -depth                3.5
Line 10:            -bed_hydraulic_conductivity     1.5e-2
Line 11:            -bed_thickness        1.6
Line 12:            -solution             4
Line 5a:       -default_solution     4
Line 6a:       165.0      3663.  # x and y location of point on river
Line 7a:            -head                 274.
Line 13:            -bottom               271.5
```

Explanation

Line 0: **RIVER** *number, description*

RIVER is the keyword for the data block.

number--positive number to designate this river. Default is 1.

description--optional character field that names the river.

Line 1: **-default_width** *default_width*

-default_width--The width of the river that is input on this line will be used for each succeeding river point that does not have an explicit definition of the river width. The default width may be redefined at any point in the definition of the river with another Line 1. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block. Optionally, **default_width**, or **-default_w**[**idth**].

default_width--Width of the river assigned by default to each subsequently defined river point.

Line 2: **-default_depth** *default_depth*

-default_depth--The depth of the river that is input on this line will be used for each succeeding river point that does not have an explicit definition of the river depth. The depth of the river is subtracted from the initial head definition for a river point to define the elevation of the bottom of the river. The default depth may be redefined at any point in the definition of the river with another Line 2. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **default_depth**, or **-default_d**[**epth**].

default_depth--Depth of the river assigned by default to each subsequently defined river point.

Line 3: **-default_bed_hydraulic_conductivity** *default_bed_hydraulic_conductivity*

-default_bed_hydraulic_conductivity--The hydraulic conductivity for the river bed that is input on this line will be used for each succeeding river point that does not have an explicit definition of hydraulic conductivity for the river bed. The default hydraulic conductivity for the river bed may be redefined at any point in the definition of the river with another Line 3. Units, L/T, are defined by the **-river_bed_hydraulic_conductivity** identifier in the **UNITS** data block. Optionally, **default_bed_hydraulic_conductivity**, **default_bed_k**, **default_k**, **-default_bed_h**[**ydraulic_conductivity**], **-default_bed_k**, or **-default_k**.

default_bed_hydraulic_conductivity--Hydraulic conductivity of the river bed assigned by default to each subsequently defined river point.

Line 4: **-default_bed_thickness** *default_bed_thickness*

-default_bed_thickness--The thickness for the river bed that is input on this line will be used for each succeeding river point that does not have an explicit definition of thickness for the river bed. The default thickness for the river bed may be redefined at any point in the definition of the river with another Line 4. Units, L, are defined by the **-river_bed_thickness** identifier in the **UNITS**

data block. Optionally, **default_bed_thickness**, **default_thickness**, **-default_bed_t**[**hickness**], or **-default_t**[**hickness**].

default_bed_thickness--Thickness of the river bed assigned by default to each subsequently defined river point.

Line 5: **-default_solution** *default_solution*

-default_solution--The solution index number input on this line will be used for each succeeding that does not have an explicit definition of a solution index number. Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. The default solution index number may be redefined at any point in the definition of the river with another Line 5. Optionally, **default_solution**, or **-default_s**[**olution**].

default_solution--A single solution index number is assigned by default to each subsequently defined river point.

Line 6: *X*, *Y*

X and *Y* location of a river point. Line 6 may be repeated as many times as needed to define the entire length of the river. At least two points must be defined. The series of points in sequence define the river by a series of line segments.

X--*X* location of a river point. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block.

Y--*Y* location of a river point. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block.

Line 7: **-head** *head*

-head--The head at the last defined *X*, *Y* river-point location. Units, L, are defined by the **-head** identifier in the **UNITS** data block. Optionally, **head**, or **-he**[**ad**].

head--Head at the river point.

Line 8: **-width** *width*

-width--The width of the river at the last defined *X*, *Y* river-point location. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block. Optionally, **width**, or **-w**[**idth**].

width--Width of the river at the river point.

Line 9: **-depth** *depth*

-depth--The depth of the river at the last defined *X*, *Y* river-point location. The depth of the river is subtracted from the initial head definition for a river point to define the elevation of the bottom of the river. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **depth**, or **-dep**[**th**].

depth--Depth of the river at the river point.

Line 10: **-bed_hydraulic_conductivity** *bed_hydraulic_conductivity*

-bed_hydraulic_conductivity--The hydraulic conductivity for the river bed at the last defined *X*, *Y* river-point location. Units, L/T, are defined by the **-river_bed_hydraulic_conductivity** identifier in the **UNITS** data block. Optionally, **bed_hydraulic_conductivity**, **bed_k**, **hydraulic_conductivity**, **k**, **-bed_h**[**ydraulic_conductivity**], **-bed_k**, **-hy**[**draulic_conductivity**], or **-k**.

bed_hydraulic_conductivity--Hydraulic conductivity of the river bed at the river point.

Line 11: **-bed_thickness** *bed_thickness*

-bed_thickness--The thickness for the river bed at the last defined *X*, *Y* river-point location. Units, L, are defined by the **-river_bed_thickness** identifier in the **UNITS** data block. Optionally, **bed_thickness**, **thickness**, **-bed_t**[**hickness**], or **-t**[**hickness**].

bed_thickness--Thickness of the river bed at the river point.

Line 12: **-solution** *solution*

-solution--The solution index number at the last defined *X*, *Y* river-point location. Solution compositions are based on index numbers that refer to solution compositions defined in the

chemical data file. Optionally, **solution**, **associated_solution**, **-s**[**olution**], or **-a**[**ssociated_solution**].

solution--Solution index number for the river point.

Line 13: **-bottom** *bottom*

-bottom--The elevation of the top of the river bed at the last defined *X*, *Y* river-point location. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **bottom**, **z**, **river_bottom**, **-bo**[**ttom**], **-z**, or **-r**[**iver_bottom**].

bottom--Elevation of the top of the river bed at the river point.

Notes

A river is defined by a sequential set of points within a **RIVER** data block that define line segments that comprise the river. A river may be defined in upstream or downstream order. Multiple rivers may be defined using multiple **RIVER** data blocks, where each river is uniquely identified by the integer following the **RIVER** keyword. The head, width, and river-bed elevation, hydraulic conductivity, and thickness must be defined at each point. For reaction-transport calculations, an associated solution must be defined for each point to specify the composition of water flowing from the river into the aquifer. The elevation of the river bottom can be defined in two ways: it can be defined explicitly with **-river_bottom** identifier, or it can be defined by defining a river depth (**-depth**) that will be subtracted from the head (**-head**) defined for the first simulation period.

Head must be explicitly defined for the first and last point in a river. For a point between the first and last point, the head may be defined explicitly (**-head**) or it can be interpolated from the last previous and next succeeding point at which head is explicitly defined. Interpolation is performed using distance along the line segments that connect the two points at which head is defined.

For each river property except head and river-bed elevation--that is, width, depth, associated solution, and river-bed hydraulic conductivity and thickness-- it is possible to define default values that will be used if no explicit value is given for a property at a river point (**-default_width**, **-default_depth**, **-default_solution**,

-default_bed_hydraulic_conductivity, **-default_bed_thickness**). The last previously defined default value for a property will be used for each point for which that property is not explicitly defined.

The flow to or from the river is determined by the head in the aquifer and the river, the elevation and area of the river bottom, and the hydraulic conductivity and thickness of the river bed. Several steps are involved in translating the series of point that define a river to the grid cells of the model. Although the sequence of steps is slightly different, the process is most easily visualized as follows. First, each pair of geographic points and their associated widths are used to define trapezoidal areas as shown in figure 1A. The collection of trapezoids has gaps between trapezoids and overlapping areas (fig. 1B). The area between the end lines of adjacent trapezoids is appended to the trapezoid that is defined first in sequence. The overlapping area is assigned to the trapezoid that is first in the sequence and overlapping areas are removed from any succeeding trapezoids (fig.1C). This removal process acts over all rivers, such that overlapping areas between rivers are assigned to the river with the smaller integer identification number. Each polygon is then intersected with each cell on the surface of the grid (fig. 1D). The area of each polygon is divided into subpolygons each of which is contained within a single grid cell (fig. 1E). Finally, the centroid of each subpolygon is calculated. Properties for a subpolygon of a river polygon are determined by interpolation of the end-point values of the line segment that defines the river polygon to the point on the line segment that is closest to the centroid (fig. 1F). If reaction-transport modeling is performed, the composition of the solution associated with the river segment is determined by proportional mixing of the end-member solution compositions defined at the endpoints of the river segment.

The difference between the algorithm described here and the actual implementation is that the removal of overlapping areas occurs after all subpolygons for all rivers have been calculated. For each cell, any intersection of the first subpolygon with a subsequent subpolygon for that cell is removed from the latter subpolygon. Then, any intersection of the second subpolygon with a subsequent subpolygon is removed, and so on.

Complete explicit or implicit definition of all points for all rivers is required for the first simulation period. Only the head (**-head**) and the solution composition (**-solution**) for a river point may be specified in subsequent simulation periods. When head or solution composition for a river point is specified in subsequent simulation

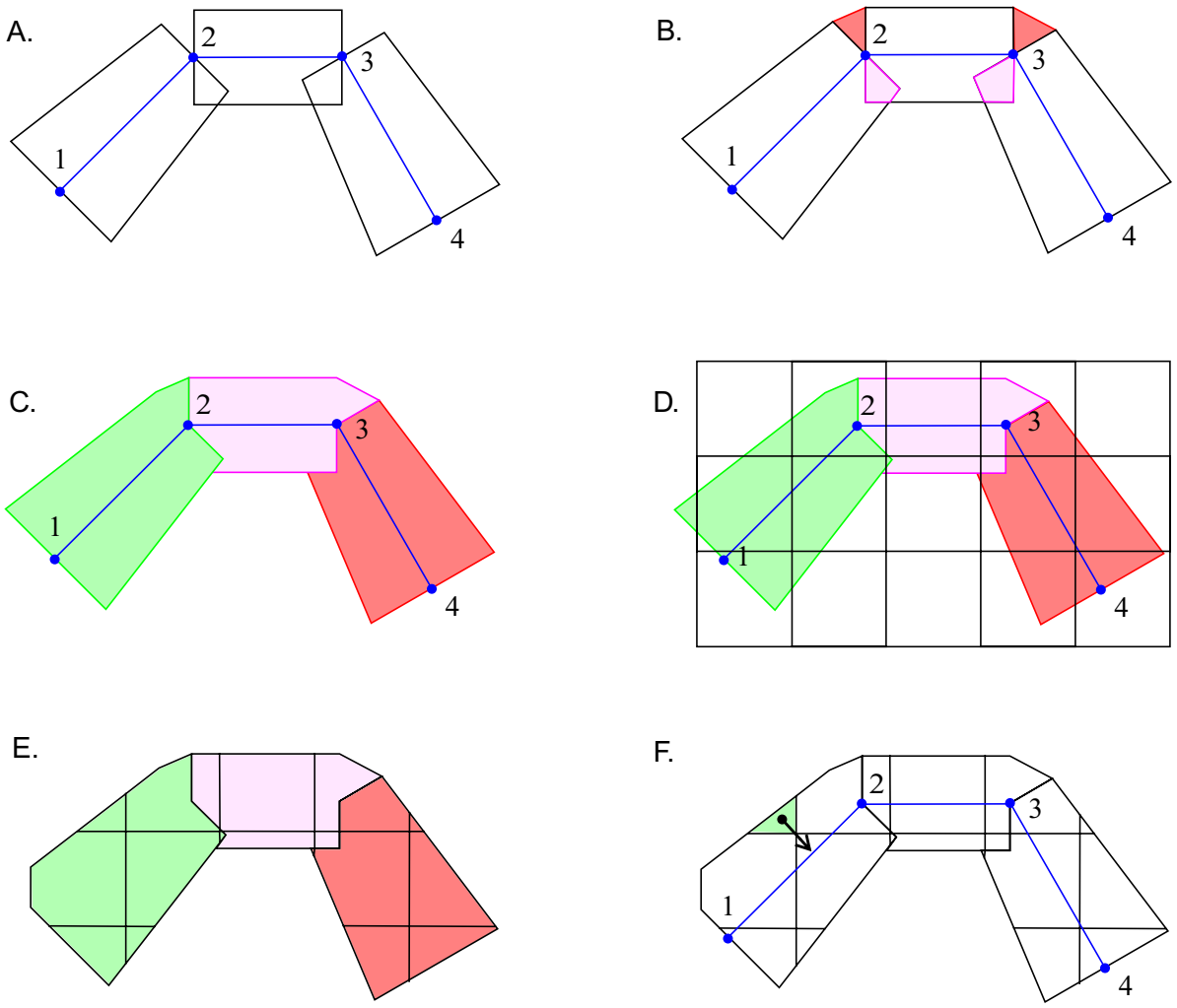

Figure 1.--Steps by which river segment areas and river segment heads are determined from the location and width data that define the river. A. Trapezoidal areas are defined by the input data. B. Gaps are filled and overlaps are eliminated. C. Revised areas of river segments. D. Intersection with grid. E. River areas for model, color indicates association with original data. F. Heads are assigned by finding point on the river nearest the centroid of the river area; head is interpolated from the end points of the river segment.

periods, the *X* and *Y* location of the point must be defined in a **RIVER** data block with the correct river number in the keyword definition. The *X* and *Y* location is followed by the **-head** and (or) **-solution** identifiers. At the beginning of each simulation period, terms that incorporate both the transient head and solution composition and the static area and river-bed definition for each river subpolygon are incorporated into the flow and transport equations.

SOLUTION_METHOD

This keyword is used to select the solver that is used to solve the flow and transport equations and to set parameters related to the solver and difference equations. If the data block is not included in the flow and transport data file, the iterative solver with default parameter values is used.

Example 1

```
Line 0:  SOLUTION_METHOD
Line 1:        -direct_solver          true
Line 2:        -space_differencing     0.5
Line 3:        -time_differencing      0.5
Line 4:        -cross_dispersion       False
```

Explanation 1

Line 0: **SOLUTION_METHOD**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-direct_solver** [***True*** *or* ***False***]

-direct_solver--Use the direct solver (D4 Gaussian elimination) for the flow and transport equations. Optionally, **direct_solver**, **direct**, or **-d**[**irect_solver**].

True *or* ***False***--**True**, use the direct solver; **false**, use the iterative solver. If neither **true** nor **false** is entered, **true** is assumed. Initial value is **false**. Optionally, **t**[**rue**] or **f**[**alse**], case independent.

Line 2: **-space_differencing** *weight*

-space_differencing--Spatial differencing for the advective term in the transport equation. Optionally, **space_differencing**, **space**, or **-sp**[**ace_differencing**].

weight--Value may range from 0.0 (upstream in space) to 0.5 (central differencing). Default 0.0.

Line 3: **-time_differencing** *weight*

-time_differencing--Time differencing for both the flow and transport equations. Optionally, **time_differencing**, **time**, or **-ti**[**me_differencing**].

weight--Value may range from 0.5 (central differencing or Crank-Nicholson) to 1.0 (fully implicit or backwards in time). Default 1.0.

Line 4: **-cross_dispersion** [***True*** *or* ***False***]

-cross_dispersion--Identifier for cross dispersive terms in the transport equations. Optionally, **cross_dispersion**, or **-c**[**ross_dispersion**].

True *or* ***False***--**True**, use cross dispersive terms in transport equations; **false**, do not include cross dispersive terms in transport equations. If neither **true** nor **false** is entered, **true** is assumed. Initial value is **false**. Optionally, **t**[**rue**] or **f**[**alse**], case independent.

Notes 1

By default, the iterative solver is used, but the direct solver is appropriate for small problems. The direct solver has no additional parameters that need to be defined. If cross dispersion terms are used (**-cross_dispersion** true), it may cause negative concentrations in the solution of the transport equations, which will be set to zero in the chemical calculations. A further result is that any resetting of concentrations to zero will cause an increase in overall mass of a solute in the system and an increase in mass-balance errors.

Example 2

```
Line 0:  SOLUTION_METHOD
Line 1:       -iterative_solver     true
Line 2:       -tolerance            1e-8
Line 3:       -save_directions      5
Line 4:       -maximum_iterations   500
Line 5:       -space_differencing   0.0
Line 6:       -time_differencing    1.0
Line 7:       -cross_dispersion     False
```

Explanation 2

Line 0: **SOLUTION_METHOD**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-iterative_solver** [***True*** *or* ***False***]

-iterative_solver--Use the restarted, generalized conjugate-gradient iterative solver for the flow and transport equations. Optionally, **iterative_solver**, **iterative**, or **-iterativ**[**e_solver**].

True** or **False--**True**, use the iterative solver; **false**, use the direct solver. If neither **true** nor **false** is entered, **true** is assumed. Initial value is **true**. Optionally, **t**[**rue**] or **f**[**alse**], case independent.

Line 2: **-tolerance** *tolerance*

-tolerance--Iterative solver has converged when the scaled residual is smaller than this number. Optionally, **tolerance**, or **-to**[**lerance**].

tolerance--Convergence criterion for the iterative solver. Default 1e-8.

Line 3: **-save_directions** *number*

-save_directions--The number of search directions between restarts of the iterative solver. Optionally, **save_directions**, **save**, or **-sa**[**ve_directions**].

number--Number of saved search directions. Default 5.

Line 4: **-maximum_iterations** *number*

-maximum_iterations--Maximum number of iterations that are allowed for the iterative solver. Optionally, **maximum_iterations**, **maximum**, **iterations**, **-m**[**aximum_iterations**], or **-iterati**[**ons**].

number--Maximum number of iterations for the iterative solver. Default 500.

Line 5: **-space_differencing** *weight*

Same as example 1, line 2.

Line 6: **-time_differencing** *weight*

Same as example 1, line 3.

Line 7: **-cross_dispersion** [***True** or **False***]

Same as example 1, line 4.

Notes 2

By default, the iterative solver is used, which is a restarted Orthomin method that solves the Schur complement reduced matrix preconditioned by triangular factorization. The identifiers, **-tolerance**, **-save_directions**, and **-maximum_iterations** apply only to the iterative solver. *Tolerance* is the limit placed on

the Euclidean norm of the relative residual that defines convergence. This parameter can not be determined *a priori*. A value for *tolerance* can be determined empirically by examining the magnitude of the changes in heads and concentrations with successively smaller values of *tolerance*.

As the number of search directions is increased, the amount of memory needed to save the vectors of the search directions increases. Normally, 5 to 10 saved search directions is adequate for convergence of the solver.

SPECIFIED_VALUE_BC

This keyword is used to define specified-value boundary conditions. For flow-only simulations, only the parameters related to flow for each cell are required. For reaction-transport calculations, the index number of a fixed solution or an associated solution is also required. This keyword data block is not needed if no specified-value boundary conditions are included in the simulation.

Example

```
Line 0:  SPECIFIED_VALUE_BC
Line 1:        -zone      0     10    10    10    10    10
Line 2:              -head            100
Line 3:              -associated_solution  6
Line 1a:       -zone      0     0     10    10    10    10
Line 4:              -fixed_solution       5
```

Explanation

Line 0: **SPECIFIED_VALUE_BC**

SPECIFIED_VALUE_BC is the keyword for the data block, no other data are included on this line.

Line 1: **-zone** x_1, y_1, z_1, x_2, y_2, z_2

-zone--A zone, which may be a single point (including 1 cell) or a 1- or 2-dimensional zone, for specified value boundary-condition definition. Optionally, **zone** or **-z**[**one**].

x_1, y_1, z_1--*X*, *Y*, and *Z* coordinate for the left (*X*), front (*Y*), lower (*Z*) corner of the rectangular zone.

x_2, y_2, z_2--*X*, *Y*, and *Z* coordinate for the right (*X*), back (*Y*), upper (*Z*) corner of the rectangular zone. Units, L, are defined by **-horizontal_grid** (*X* and *Y* coordinates) and **-vertical_grid** (*Z* coordinates) identifiers in the **UNITS** data block.

Line 2: **-head** *property*

-head--Head at the boundary. Optionally, **head** or **-he**[**ad**].

property--The head (L) at the boundary may be entered with any of the four methods for defining a spatially distributed property: (1) a single value for the zone, (2) **by_cell** followed by a value for each cell in the zone, (3) **file** followed by a file name, a value for each cell in the zone is read

from the file, or (4) **X**, **Y**, or **Z**, $value_1$, $distance_1$, $value_2$, $distance_2$, a value is linearly interpolated from the end-point values defined by $value_1$ and $value_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for locations outside the range of $distance_1$ to $distance_2$, the value at the nearest end point ($distance_1$ or $distance_2$) is used. Units, L, are defined by the **-head** identifier in the **UNITS** data block.

Line 3: **-associated_solution** *property*

-associated_solution--Index numbers of the associated solutions. Inflowing solutions at the boundary will have this composition, outflowing solutions have the same composition as the solution in the cell. Optionally, **associated_solution**, **associated**, or **-a**[**ssociated_solution**].

property--Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Solutions or mixtures of solutions may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 3 illustrates method 1.

Line 4: **-fixed_solution** *property*

-fixed_solution--Index numbers for fixed solution compositions at the boundary. Optionally, **fixed_solution**, **fixed**, or **-f**[**ixed_solution**].

property--Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Solutions or mixtures of solutions may be entered for the zone with any of the four methods for defining a spatially distributed property: (1) a single index, the composition represented by the index is applied to the entire zone, (2) **by_cell** followed by an index for each cell in the zone, an index is explicitly defined for each cell, (3) **file** followed by a file name, an index for each cell in the zone is read from the file, or (4) **X**, **Y**, or **Z**, $index_1$, $distance_1$, $index_2$, $distance_2$, a composition is linearly interpolated from the end-member compositions defined by indices $index_1$ and $index_2$ for each cell in the zone; interpolation is proportional to the fractional distance of the node location between $distance_1$ and $distance_2$ along the given coordinate direction; for distances outside the range of $distance_1$ to $distance_2$, the composition at the nearest end point ($distance_1$ or $distance_2$) is used. Line 4 illustrates method 1.

Notes

The zones for all boundary conditions must be 0-, 1-, or 2-dimensional.

Two identifiers define different boundary conditions for the concentrations of chemical components at the boundary, **-fixed_solution** and **-associated_solution**. With the **-fixed_solution** identifier, the concentrations of each component in the boundary cell are constant for the simulation period and equal to the values in the solution with the specified index number (the solution composition is defined in the chemical data file). With the **-associated_solution** identifier, the concentrations of the components in the boundary cell may vary as determined by reaction and transport. If flow leaves the domain during a time step, the concentrations in the fluid that leaves are equal to the concentrations in the boundary cell. If flow enters the domain during a time step, the concentrations

in the fluid that enters are equal to the values in the solution with the specified index number (the solution composition is defined in the chemical data file).

Multiple zones may be used within the **FLUX_BC**, **LEAKY_BC**, and **SPECIFIED_VALUE_BC** data blocks to define boundary conditions within the domain. Different boundary conditions for a single cell (node) may be defined multiple times as part of different zone definitions and different keyword data blocks. The boundary condition that is used for a cell is the last zone definition that defines a boundary condition for that cell.

STEADY_FLOW

This keyword is used to specify that the flow is steady state and to define parameters necessary to attain a steady-state flow condition. If the simulation is defined to have steady flow, then an initial flow-only calculation is performed to determine the steady-state velocities and these velocities are used in the transport equations throughout the simulation.

Example

```
Line 0:  STEADY_FLOW true
Line 1:        -head_tolerance              1e-8
Line 2:        -flow_balance_tolerance      0.001
Line 3:        -minimum_time_step           1
Line 4:        -maximum_time_step           1000
Line 5:        -head_change_limit           100.
```

Explanation

Line 0: **STEADY_FLOW**

STEADY_FLOW specifies that steady-state flow conditions will be calculated. If the simulation is a reactive transport simulation (**FLOW_ONLY** false), then the steady-state velocities will be used for transport calculations throughout the run. If a **STEADY_FLOW** data block is not included in the flow and transport data file, the simulation is assumed not to be steady flow.

[***True*** *or* ***False***]--a value of **true** (optionally, **t**[**rue**]) indicates that an initial calculation will be performed to calculate steady-state velocities and these will be used for all transport calculations. A value of **false** (optionally, **f**[**alse**]) indicates that new velocities will be calculated at each time step of the simulation. If neither **true** nor **false** are entered, **true** is assumed.

Line 1: **-head_tolerance** *tolerance*

-head_tolerance--Tolerance for determining steady-state head is entered on this line. Optionally, **head_tolerance**, **head_tol**, or **-h**[**ead_tolerance**].

tolerance--If head changes over an iteration are less than this tolerance, the head field is assumed to have reached steady state. Units, L, are defined by **-head** identifier in the **UNITS** data block. Default, 10^{-8}.

Line 2: **-flow_balance_tolerance** *tolerance*

-flow_balance_tolerance--Tolerance for determining steady-state flow balance is entered on this line. Optionally, **flow_tol**, **flow_tolerance**, **flow_balance_tol**, **flow_balance_tolerance**, **flow_t**[**olerance**], or **-flow_b**[**alance_tolerance**].

tolerance--If flow balance is less than this tolerance, the flow field is assumed to have reached steady state. Units are dimensionless fraction of total flow. Default, 0.001.

Line 3: **-minimum_time_step** *time_step*

-minimum_time_step--Initial time step for time stepping to steady-state flow; also the minimum time step used in the steady-state flow calculation. Optionally, **minimum**, **minimum_time**, **minimum_time_step**, or **mi**[**nimum_time_step**].

time_step--Initial and minimum time step for steady-flow calculation. Units, T, are defined by **-time** identifier in the **UNITS** data block. Default, value for **-time_step** defined in **TIME_CONTROL**.

Line 4: **-maximum_time_step** *time_step*

-maximum_time_step--Maximum time step used in the steady-state flow calculation. Optionally, **maximum**, **maximum_time**, **maximum_time_step**, or **ma**[**ximum_time_step**].

time_step--Maximum time step for steady flow calculation. Default, 1000 times the value for **-time_step** defined in **TIME_CONTROL**.

Line 5: **-head_change_limit** *limit*

-head_change_limit--Target maximum head change allowed for a single time step is entered on this line. Optionally, **head_change**, **head_change_limit**, or **-head_c**[**hange_limit**].

limit--The time step is adjusted during the steady-state calculation to attempt to limit head changes during an iteration to less than *limit*. Units, L, are defined by **-head** identifier in the **UNITS** data block. Default, 0.3 times the thickness of the model domain converted to head units.

Notes

The **STEADY_FLOW** true can be used with or without the **FLOW_ONLY** true. If **STEADY_FLOW** is true, an additional flow-only calculation is performed at the beginning of the PHAST calculation to determine the steady-state head and velocity fields. The calculation ensures that a velocity field is calculated such that heads are constant within the tolerance specified by **-head_tolerance** and flow balance is satisfied within the fraction of total inflow and outflow specified by **-flow_balance_tolerance**. The steady-state velocity field is used in all transport calculations for the remainder of the run. The **STEADY_FLOW** true option saves CPU time by eliminating the calculation of a new velocity field for each time step, but allows no transient changes in head or flow velocities.

TIME_CONTROL

This keyword is used to define the time step and the time at which the current simulation period ends. The time of the end of each simulation period is cumulative from the beginning of the run. The data block is mandatory for all simulations.

Example

```
Line 0:  TIME_CONTROL
Line 1:        -time_step       10    day
Line 2:        -time_change     1     yr
```

Explanation

Line 0: **TIME_CONTROL**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-time_step** *length,* [*units*]

-time_step--Time-step length. Optionally, **step**, **time_step**, **delta**, **delta_time**, **-s**[**tep**], **-time_s**[**tep**], or **-d**[**elta_time**].

length--Time-step length (T), units are defined with *units*. Default units are defined by **-time** identifier in **UNITS** data block.

units--Units for time step may be may be “seconds”, “minutes”, “hours”, “days”, or “years” or an abbreviation of one of these units. Default *units* are defined by **-time** identifier in **UNITS** data block.

Line 1: **-time_change** *time,* [*units*]

-time_change--Time at which the simulation period ends. This time may be the end of the run or the time at which a new simulation period is begun. Optionally, **time_change**, **change_time**, **end**, **end_time**, **time_end**, **-time_c**[**hange**], **-c**[**hange_time**],**-e**[**nd_time**], or **-time_e**[**nd**].

time--Time at which the simulation period ends, which is the cumulative time since the beginning of the run. *Time* (T) units are defined with *units*. Default units are defined by **-time** identifier in **UNITS** data block.

units--Units for *time* may be may be “seconds”, “minutes”, “hours”, “days”, or “years” or an abbreviation of one of these units. Default *units* are defined by **-time** identifier in **UNITS** data block.

Notes

The **TIME_CONTROL** data block, including both identifiers, is required for each simulation period. The default unit for the **-time_step** and **-time_change** data are given by the **-time** identifier in **UNITS** data block. However, if units are specified in the **TIME_CONTROL** data block, then the specified units override the default unit. The units specified for **-time_step** and **-time_change** need not be the same, if the units are explicitly defined.

TITLE

This keyword data block is used to include two title lines for a simulation in each of the output files, except the files *prefix*.**xyz.**xxx.

Example

```
Line 0:  TITLE
Line 1:  line 1
Line 1a: line 2
```

Explanation

Line 0: **TITLE**

TITLE is the keyword for the data block.

Line 1: *title*

title--The title may continue on as many lines as desired, however, only two lines are included in each of the output files, except *selected_output*. Lines are read and saved as part of the title until a keyword begins a line or until the end of the flow and transport data file.

Notes

The **TITLE** data block is intended to be used to identify the simulation in the output files, except *selected_output* (The file name *selected_output* is defined in the **SELECTED_OUTPUT** data block in the chemical data file). Be careful not to begin a line of the title with a keyword because that signals the end of the **TITLE** data block. More than two lines may be entered, but only the first two are included in the output files. If more than one title keyword is entered in the flow and transport data file, only the lines from the last **TITLE** data block will appear in the output files.

UNITS

This keyword data block is used to specify the units for the input data. All flow and transport calculations within PHAST use SI units. The **UNITS** data block provides the information necessary to convert the input data from the input units to SI units. However, output units are always SI except for time, which is the unit as given in the **-time** identifier of this keyword data block. The **UNITS** data block is mandatory for all calculations and must be present in the data for the first simulation period.

Example

```
Line 0:  UNITS
Line 1:       -time                        years
Line 2:       -horizontal_grid             km
Line 3:       -vertical_grid               ft
Line 4:       -head                        ft
Line 5:       -hydraulic_conductivity      m/d
Line 6:       -specific_storage            1/ft
Line 7:       -dispersivity                m
Line 8:       -flux                        m/s
Line 9:       -leaky_hydraulic_conductivity    km/yr
Line 10:      -leaky_thickness             km
Line 11:      -well_diameter               in
Line 12:      -well_flow_rate              gpm
Line 13:      -river_bed_hydraulic_conductivity     m/s
Line 14:      -river_bed_thickness         m
```

Explanation

Line 0: **UNITS**

Keyword for the data block. No other data are input on the keyword line.

Line 1: **-time** *time_units*

-time--Default units for time for input data and units of time that are used for output data.

time_units--Units for time (T) may be “seconds”, “minutes”, “hours”, “days”, or “years” or an abbreviation of one of these units. Optionally, **time** or **-t**[**ime**].

Line 2: **-horizontal_grid** *units*

-horizontal_grid--Units of input data used to define *X* and *Y* coordinate direction distances in the **GRID** data block and all **-zone** definitions. Optionally, **horizontal_grid**, **horizontal**, or **-ho**[**rizontal_grid**].

units--Units for horizontal distance in the *X* and *Y* coordinate directions (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units.

Line 3: **-vertical_grid** *units*

-vertical_grid--Units of input data used to define *Z* coordinate direction distances in the **GRID** data block and all **-zone** definitions. Optionally, **vertical_grid**, **vertical**, or **-v**[**ertical_grid**].

units--Units for vertical distance (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units.

Line 4: **-head** *units*

-head--Units of input data used for head in any keyword data block. Optionally, **head** or **-he**[**ad**].

units--Units for head (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units.

Line 5: **-hydraulic_conductivity** *units*

-hydraulic_conductivity--Units of input data for hydraulic conductivity in the **MEDIA** data block. Optionally, **hydraulic_conductivity**, **K**, **-hy**[**draulic_conductivity**], or **-K**.

units--Units for hydraulic conductivity (L/T). Length (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units. Time (T) may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Units must include the divide symbol, "/", and must contain no spaces.

Line 6: **-specific_storage** *units*

-specific_storage--Units of input data for specific storage in the **MEDIA** data block. Optionally, **specific_storage**, **storage**, **-s**[**pecific_storage**], or **-s**[**torage**].

units--Units for specific storage (1/L). Length (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units. Units must include the "1/", and must contain no spaces.

Line 7: **-dispersivity** *units*

-dispersivity--Units of input data for dispersivity in the **MEDIA** data block. Optionally, **dispersivity**, **alpha**, **-d**[**ispersivity**], or **-a**[**lpha**].

units--Units for hydraulic conductivity (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units.

Line 8: **-flux** *units*

-flux--Units of input data for fluid flux in the **FLUX_BC** data blocks. Optionally, **flux** or **-f**[**lux**].

units--Units for fluid flux (L/T). Length (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units. Time (T) may be "seconds", "minutes", "hours", "days", or "years" or an abbreviation of one of these units. Units must include the divide symbol, "/", and must contain no spaces.

Line 9: **-leaky_hydraulic_conductivity** *units*

-leaky_hydraulic_conductivity--Units of input data for hydraulic conductivity in the **LEAKY_BC** data blocks. Optionally, **leaky_hydraulic_conductivity**, **leaky_K**, **-leaky_h**[**ydraulic_conductivity**], or **-leaky_K**.

units--Units for hydraulic conductivity (L/T). Length (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units. Time (T) may be "seconds", "minutes", "hours", "days", or "years" or an

abbreviation of one of these units. Units must include the divide symbol, "/", and must contain no spaces.

Line 10: **-leaky_thickness** *units*

-leaky_thickness--Units of input data for thickness of the leaky boundary in the **LEAKY_BC** data blocks. Optionally, **leaky_thickness**, **thickness**, **-leaky_t**[**hickness**], or **-t**[**hickness**].

units--Units for thickness of the leaky boundary (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of these units.

Line 11: **-well_diameter** *units*

-well_diameter--Units of input data for diameter or radius of wells in **WELL** data blocks. Optionally, **well_diameter** or **-well_d**[**iameter**].

units--Units for well diameter or radius, L, may be English (for example "inches" or "feet") or metric (for example "millimeters", "centimeters", or "meters") or an abbreviation of one of these units.

Line 12: **-well_flow_rate** *units*

-well_flow_rate--Units of input data for flow rates in **WELL** data blocks. Optionally, **well_flow_rate**, **well_pumpage**, **-well_f**[**low_rate**], or **-well_p**[**umpage**].

units--Units for flow rate (L3/T). Units may be English (for example, "gal/min", "gpm", "ft^3/sec"), or metric (for example, "Liter/minute", or "meters3/day") or an abbreviation of one of these units.

Line 13: **-river_bed_hydraulic_conductivity** *units*

-river_bed_hydraulic_conductivity--Units of input data for hydraulic conductivity in **RIVER** data blocks. Optionally, **river_bed_hydraulic_conductivity**, **river_bed_k**, **river_k**, **-river_bed_h**[**ydraulic_conductivity**], **-river_bed_k**, or **-river_k**.

units--Units for hydraulic conductivity (L/T). Length (L) may be English ("inches", "feet", or "miles") or metric ("millimeters", "centimeters", "meters", or "kilometers") or an abbreviation of one of

these units. Time (T) may be “seconds”, “minutes”, “hours”, “days”, or “years” or an abbreviation of one of these units. Units must include the divide symbol, “/”, and must contain no spaces.

Line 14: **-river_bed_thickness** *units*

-river_bed_thickness--Units of input data for thickness of the river bed in RIVER data blocks. Optionally, **river_bed_thickness**, **river_thickness**, **-river_bed_t**[**hickness**], or **-river_t**[**hickness**].

units--Units for thickness of the river bed (L) may be English (for example, “inches”, “feet”, or “miles”) or metric (for example, “millimeters”, “centimeters”, “meters”, or “kilometers”) or an abbreviation of one of these units.

Notes

The **UNITS** data block must be present in the data for the first simulation period for all flow and transport calculations. The definitions in this data block apply for all processing of the flow and transport data file. Note that all definitions, except **-time**, apply only to the input data. All output data is in SI units, except that the time unit is that defined with the **-time** identifier.

WELL

This keyword is used to define an injection or pumping well. Multiple **WELL** data blocks may be used to define all of the wells in the domain.

Example

```
Line 0:  WELL 122 Metropolis Injection Well 122
Line 1:       1766.      2356.
Line 2:       -diameter             12
Line 3:       -injection_rate  4.5
Line 4:       -solution        16
Line 5a:      -elevation       101.  107.
Line 5b:      -elevation       143.  153.
Line 5c:      -elevation       175.  183.
Line 0:  WELL 165 Metropolis Supply Well 165
Line 1:       1833.      2320.
Line 6:       -radius               12
Line 7:       -pumping_rate    4.5
Line 8:       -land_surface_datum   292.
Line 9a:      -depth           42.   47.
Line 9b:      -depth           99.   103.
Line 10:      -allocate_by_pressure_and_mobility    true
```

Explanation

Line 0: **WELL** *number, description*

WELL is the keyword for the data block.

number--positive number to designate this well. Default is 1.

description--optional character field that names the well.

Line 1: *X, Y*

X--*X* location of the well. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block.

Y--*Y* location of the well. Units, L, are defined by the **-horizontal_grid** identifier in the **UNITS** data block.

Line 2: **-diameter** *diameter*

-diameter--The diameter of the well. Units, L, are defined by the **-well_diameter** identifier in the **UNITS** data block. Optionally, **diameter**, or **-di**[**ameter**].

diameter--Diameter of the well.

Line 3: **-injection_rate** *injection_rate*

-injection_rate--The rate of fluid injection into the well. Units, L^3/T, are defined by the **-well_flow_rate** identifier in the **UNITS** data block. Optionally, **injection_rate**, **injection**, or **-i**[**njection_rate**].

injection_rate--Rate of fluid injection into the well.

Line 4: **-solution** *solution*

-solution--The solution index number that defines composition of water injected into the well. Identifier is required for each injection well. Solution compositions are based on index numbers that refer to solution compositions defined in the chemical data file. Optionally, **solution**, **associated_solution**, **-s**[**olution**], or **-a**[**ssociated_solution**].

solution--Solution index number for the river point.

Line 5: **-elevation** $elevation_1$, $elevation_2$

-elevation--The top and bottom elevations of an open interval in the well are defined. Multiple **-elevation** identifiers are used to define all open intervals for the well. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **elevation**, **elevations**, or **-e**[**levations**].

$elevation_1$--Elevation of start of open interval in the well.

$elevation_2$--Elevation of end of open interval in the well.

Line 6: **-radius** *radius*

-radius--The radius of the well. Units, L, are defined by the **-well_diameter** identifier in the **UNITS** data block. Optionally, **radius**, or **-r**[**adius**].

radius--Radius of the well.

Line 7: **-pumping_rate** *pumping_rate*

-pumping_rate--The rate water is removed from the well. Units, L^3/T, are defined by the **-well_flow_rate** identifier in the **UNITS** data block. Optionally, **pumping_rate**, **pumping**, **pumpage**, **-pu**[**mping_rate**], or **-pu**[**mpage**].

pumping_rate--Rate water is removed from the well.

Line 8: **-land_surface_datum** *land_surface_datum*

-land_surface_datum--The elevation of the land surface at the well. This data item is required only if open intervals are defined with the **-depth** identifier. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **land_surface_datum**, **lsd**, **-l**[**and_surface_datum**], or **-l**[**sd**].

land_surface_datum--Elevation of land surface at the well.

Line 9: **-depth** $depth_1$, $depth_2$

-depth--The top and bottom of an open interval in the well are defined by depths below land surface. The **-land_surface_datum** identifier must be used to define the elevation at the top of the well. Multiple **-depth** identifiers are used to define all open intervals for the well. Units, L, are defined by the **-vertical_grid** identifier in the **UNITS** data block. Optionally, **depth**, **depths**, or **-de**[**pths**].

$depth_1$--Depth of start of open interval in the well.

$depth_2$--Depth of end of open interval in the well.

Line 10: **-allocate_by_pressure_and_mobility** [***True*** *or* ***False***]

-allocate_by_pressure_and_mobility--If true, well-bore flow-rate allocation is by the product of mobility and pressure difference; if false, well-bore flow-rate allocation is by mobility only (Kipp, 1987, p. 34 and p. 122). Optionally, **allocation_by_pressure_and_mobility**, **allocate_by_pressure_and_mobility**, **pressure_and_mobility**,

-al[**location_by_pressure_and_mobility**], **-al**[**location_by_pressure_and_mobility**], or **-pr**[**essure_and_mobility**].

[***True*** *or* ***False***]--**True** allocates flow by mobility and pressure difference, **false** allocates flow by mobility only. Default is **true**, if neither ***True*** nor ***False*** is entered; initial setting is **false**.

Notes

Multiple wells may be defined using multiple **WELL** data blocks, where each well is uniquely identified by the integer following the **WELL** keyword. The *X*-*Y* location, diameter (or radius), flow rate, and open intervals must be defined for each well. If the well is an injection well, the flow rate is defined with the **-injection_rate** identifier as a positive number and an associated solution composition (**-solution** identifier) must be entered for reaction-transport calculations. If the well is a pumping well, the flow rate is defined with the **-pumping_rate** identifier as a positive number. Internal to the program, pumping rates are negative and injection rates are positive. A well may be changed from an injection well to a pumping well (or the reverse) in different simulation periods. The elevation of open intervals in a well can be defined in two ways: (1) the elevations of the top and bottom of interval can be defined explicitly with the **-elevation** identifier, or (2) the land surface elevation can be defined with **-land_surface_datum** and the depths to the top and bottom of each open interval can be defined with the **-depth** identifier; the depths are subtracted from the land-surface datum to determine the elevations of the open intervals.

By default, allocation of well-bore flow to the open intervals is by a formula that depends only on mobility. Alternatively, the **-allocate_by_pressure_and_mobility** can be set to true to cause the allocation of well-bore flow to be determined by the product of mobility and pressure difference between the well and the aquifer.

Complete definitions of all wells are required for the first simulation period. Only the flow rate (**-injection_rate** or **-pumping_rate**) and the solution composition (**-solution**) for a well may be specified in subsequent simulation periods. When flow rate or solution composition for a well are specified in subsequent simulation periods, a **WELL** data block is needed with the correct well number in the keyword definition. The keyword is followed by an identifier for the flow rate (**-injection_rate** or **-pumping_rate**) and (or) the **-solution** identifier.